

Федеральное государственное автономное  
образовательное учреждение  
высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»  
Институт космических и информационных технологий  
Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ  
Заведующий кафедрой  
\_\_\_\_\_ Г. М. Цибульский  
подпись  
« \_\_\_\_\_ » \_\_\_\_\_ 2018 г.

**БАКАЛАВРСКАЯ РАБОТА**

09.03.02 — Информационные системы и технологии  
Разработка модуля устранения шумов в задачах постсегментации спутниковых  
изображений

Руководитель	_____	доцент, канд. техн. наук	Р. В. Брежнев
	подпись, дата		
Выпускник	_____		С. В. Падалко
	подпись, дата		

Красноярск 2018

Продолжение титульного листа бакалаврской работы по теме «Разработка модуля устранения шумов в задачах постсегментации спутниковых изображений»

Нормоконтроль

\_\_\_\_\_  
подпись, дата

Р. В. Брежнев

Федеральное государственное автономное  
образовательное учреждение  
высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»  
Институт космических и информационных технологий  
Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ  
Заведующий кафедрой  
\_\_\_\_\_ Г. М. Цибульский  
подпись  
« \_\_\_\_\_ » \_\_\_\_\_ 2018 г.

**ЗАДАНИЕ  
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ  
в форме бакалаврской работы**

Студенту Падалко Сергею Витальевичу  
Группа КИ14-11Б, направление 09.03.02 «Информационные системы и технологии», профиль 09.03.02.04 «Информационные системы и технологии в медиаиндустрии».

Тема выпускной квалификационной работы «Разработка модуля устранения шумов в задачах постсегментации спутниковых изображений».

Утверждена приказом по университету № 4533/с от 29.03.2018

Руководитель ВКР Р.В. Брежнев старший преподаватель кафедры систем искусственного интеллекта ИКИТ СФУ.

Исходные данные для ВКР: задание на бакалаврскую работу, полученное в рамках научно-учебной лаборатории «Информационной поддержки космического мониторинга» Института космических и информационных технологий.

Перечень разделов ВКР:

- введение;
- анализ систем фильтрации изображений;
- выводы по главе 1;
- практическая часть;
- заключение;
- список использованных источников;
- приложения А — Г (программный код фильтра Гаусса, программный код медианного фильтра, плакаты презентации, отчет «Антиплагиат»).

Перечень графического материала: презентация «Разработка модуля устранения шумов в задачах постсегментации спутниковых изображений».

Руководитель ВКР

\_\_\_\_\_  
подпись

Р. В. Брежнев

Задание принял к исполнению

\_\_\_\_\_  
подпись

С. В. Падалко

«\_\_» \_\_\_\_\_ 2018 г.

## График

выполнения выпускной квалификационной работы студентом направления 09.03.02 «Информационные системы и технологии», профиля 09.03.02.04 «Информационные системы и технологии в медиаиндустрии».

График выполнения выпускной квалификационной работы приведен в таблице 1.

Таблица 1 — График выполнения этапов ВКР

Наименование этапа	Срок выполнения этапа	Результат выполнения этапов	Примечание руководителя (отметка о выполнении этапа)
Ознакомление с целью и задачами работы	07.03-13.03	Краткое эссе по теме ВКР	Выполнено
Сбор литературных источников	14.03-20.03	Список источников литературы	Выполнено
Анализ собранных литературных источников	21.03-27.03	Реферат о проблемно-предметной области	Выполнено
Уточнение и обоснование актуальности цели и задач ВКР	28.03-3.04	Окончательная формулировка цели и задач ВКР	Выполнено
Решение первой задачи ВКР	4.04-10.04	Доклад и презентация по первой задаче ВКР	Выполнено
Решение второй задачи ВКР	11.04-17.04	Доклад и презентация по второй задаче ВКР	Выполнено

## Окончание таблицы 1

Наименование этапа	Срок выполнения этапа	Результат выполнения этапов	Примечание руководителя (отметка о выполнении этапа)
Решение третьей задачи ВКР	18.04-24.04	Доклад и презентация по третьей задаче ВКР	Выполнено
Подготовка доклада и презентации по теме ВКР	25.04-08.05	Доклад с презентацией на тему ВКР	Выполнено
Компоновка отчета по результатам решения задач ВКР	09.05-31.05	Отчет по результатам решения задач ВКР	Выполнено
Первичный нормоконтроль (Н/К)	06.06	Пояснительная записка, презентация к ВКР	Выполнено
Предварительная защита результатов ВКР	07.06	Доклад и презентация о проделанной работе	Выполнено
Вторичный нормоконтроль (Н/К)	15.06	Пояснительная записка, презентация к ВКР	Выполнено
Итоговый нормоконтроль (Н/К)	17.06	Пояснительная записка, презентация к ВКР	Выполнено
Защита ВКР	21.06	Доклад и презентация по результатам бакалаврской работы	Выполнено

Руководитель ВКР

Студент гр. КИ14-116

\_\_\_\_\_

подпись

\_\_\_\_\_

подпись

Р. В. Брежнев

С. В. Падалко

## СОДЕРЖАНИЕ

Введение.....	4
Глава 1. Анализ систем фильтрации изображений.....	6
1.1 Предварительная обработка изображений.....	6
1.2 Анализ фильтров изображений.....	8
1.3 Оценка качества изображения.....	14
1.4 Вывод по первой главе.....	18
Глава 2. Практическая часть .....	19
2.1 Диаграмма вариантов использования .....	19
2.2 Диаграмма деятельности.....	20
2.3 Диаграмма развертывания .....	25
2.4 Разработка модуля QGIS .....	26
2.4.1. Установка необходимых модулей QGIS.....	26
2.4.2. Разработка интерфейса модуля.....	26
2.4.3. Разработка программного кода.....	27
2.5 Программная реализация фильтра Гаусса.....	28
2.6 Программная реализация медианного фильтра.....	30
2.7 Результаты работы модулей .....	31
Заключение .....	35
Список использованных источников .....	36
Приложение А Программный код фильтра Гаусса .....	38
Приложение Б Программный код медианного фильтра .....	40
Приложение В Плакаты презентации .....	42
Приложение Г Отчет «Антиплагиат» .....	50

## ВВЕДЕНИЕ

В настоящее время орбита Земли окружена спутниками дистанционного зондирования Земли различного назначения. Они оборудованы особыми сенсорными системами, которые способны улавливать отраженный от поверхности Земли свет и другие виды излучений. Данные сенсоры не работают безостановочно, включаясь периодически. Каждый сенсор настроен на определенную длину волны и фиксирует лишь эти излучения. После чего данные, полученные с сенсоров, объединяются, происходит оцифровка и отправка на станции, предназначенные для приема.

Новейшие спутники дистанционного зондирования Земли являются сложными техническими устройствами, от которых требуется регулярно, четко и не отступая от алгоритмов выполнять команды по съемке объектов и районов земной поверхности. Главной частью подобных спутников является приемная аппаратура, которая отвечает за наведение, фокусирование, съемку и передачу на Землю изображений в различных спектральных диапазонах.

Спутниковые изображения на данный момент применяются в самых разных отраслях деятельности человека. Таких как сельское хозяйство, геологические и гидрологические исследования, лесоводство, охрана природы, планирование территорий, образование, разведка и военное дело. Изображения могут быть как в видимой части спектра, так и в ультрафиолетовой, инфракрасной и других частях диапазона. Также существуют различные карты рельефа, которые были получены с помощью радарной съемки [1].

В сельскохозяйственной отрасли с использованием спутниковых изображений решается целый ряд тематических задач: оценка состояния и использования полей севооборотов, параметров плодородия почв и их деградации, состояния растительного покрова на пашне, залежах, сенокосных и пастбищных угодьях, прогнозирование урожайности и др.

Объектом агромониторинга являются земли сельскохозяйственного назначения, для которых заданы требования к форме и пространственной



структуре. Изменения, наблюдаемые на спутниковых изображениях данных объектов, связаны с естественным ходом вегетации, погодными условиями и проводимыми агротехническими мероприятиями.

Спутниковые изображения в исходном состоянии использовать неэффективно, так как присутствует большое количество лишних данных, таких как незначительные участки, которые можно опустить при анализе. Именно поэтому используется предварительная обработка изображений. В нее входит фильтрация, в частности медианный фильтр и фильтр Гаусса. Они позволяют объединить близлежащие области в одну и подавить маленькие области, которые находятся вдали от больших.

Цель работы: разработка программного модуля, реализующего фильтр Гаусса и медианный фильтр для применения их к спутниковым изображениям.

Для достижения поставленной цели потребовалось решить следующие задачи:

- обзор методов и алгоритмов фильтрации изображений;
- разработка программного модуля, реализующего алгоритм фильтра Гаусса;
- разработка программного модуля, реализующего алгоритм медианного фильтра.

## **Глава 1. Анализ систем фильтрации изображений**

### **1.1 Предварительная обработка изображений**

Процесс обработки изображений состоит из нескольких этапов. Самым важным является предварительная обработка изображений. Данный этап отвечает за фильтрацию шумов, помех и прочих недостатков на изображении. Задачи предварительной обработки изображений следующие:

- подавление шумов;
- коррекция цветов;
- изменение диапазона яркостей;
- коррекция геометрических искажений;
- изменение резкости [2].

Главной задачей предварительной обработки изображения является улучшение качества изображения. Методы предварительной обработки разнообразны и зависят от задачи исследований и от того, каким путем получено исходное изображение.

Шумоподавление изображений, в основном, служит для улучшения визуального восприятия. Но помимо этого, шумоподавление применяют и в практических целях, например в медицине, увеличивая четкость изображения на рентгеновских снимках.

В многоканальном изображении шум может иметь разную интенсивность для разных каналов. В результате этого шум может иметь разные оттенки. Наиболее заметен шум на однотонных участках. Чем темнее участок, тем лучше его видно [3].

Источниками шумов могут быть разные вещи:

- условия съемки;
- искажение данных при повреждении информации на носителе;
- шум переноса заряда;
- шум квантования АЦП;

- тепловой шум матрицы;
- помехи при передаче по аналоговым каналам линии передачи.

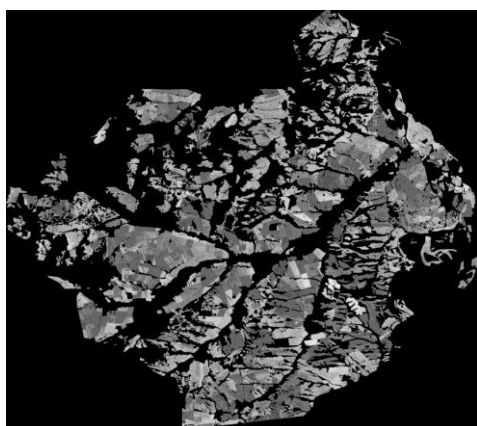
Шумы также поддаются разделению на виды. По типу искажения шумы возможно разделить на:

- аддитивный шум, возникающий из-за аппаратуры;
- мультипликативный шум, возникающий из-за возможных изменений параметров канала передачи;
- импульсный шум, возникающий из-за проблем с программным обеспечением;

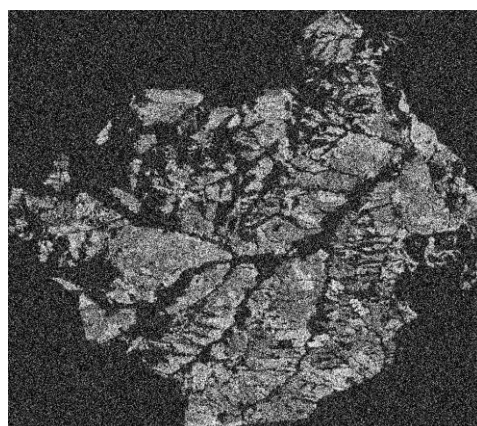
Если воспользоваться визуальным подходом, то шумы можно классифицировать как:

- белый шум – пиксели случайны и не коррелированы друг с другом;
- импульсный шум – характеризуется заменой значений некоторых пикселей на изображении на случайные или фиксированные значения;
- «битые пиксели» - пиксели или области пикселей, имеющие случайные значения, чаще всего связанные с дефектом аппаратуры;
- яркостные пятна – характерны для аналогового сигнала;
- царапины – возникающие из-за физического повреждения пленки или недостатков на объективе [4].

На рисунке 1 представлено изображение, подверженное белому шуму.



а



б

Рисунок 1 — Зашумленное изображение: а – незашумленное изображение; б – зашумленное изображение (белый шум)

На данный момент отсутствуют универсальные алгоритмы, способные определять и подавлять все имеющиеся на изображении виды шумов. Чаще всего фильтры направлены на определенный вид шума. Большую часть шумов можно представить как разновидность белого шума, именно поэтому разработчики делают приоритет на подавление данного вида [5].

## **1.2 Анализ фильтров изображений**

Самым распространенным способом борьбы с шумами является фильтрация. Под фильтрацией изображений понимают операцию, имеющую своим результатом изображение того же размера, полученное из исходного по некоторым правилам. Обычно интенсивность (цвет) каждого пикселя результирующего изображения обусловлена интенсивностями (цветами) пикселей, расположенных в некоторой его окрестности в исходном изображении.

Правила, задающие фильтрацию (их называют фильтрами), могут быть самыми разнообразными. Стоит отметить, что операция, заключающаяся в последовательном применении двух или более фильтров, тоже является фильтрацией. Исходя из этого, можно сказать, что составная фильтрация включает в себя несколько простых [6].

Так как изображение можно считать двумерным массивом, именно поэтому для фильтрации используются двумерные фильтры, которые чаще всего представляют собой квадратные матрицы. Каждый элемент матрицы является числом, которое можно назвать коэффициентом.

Алгоритмы фильтрации в основном одинаковые. Они представляют собой проход каждого пикселя слева-направо, сверху-вниз. Происходит перемножение значений исходного изображения и соответствующего коэффициента. Происходит суммирование полученных значений, а после деление, на коэффициент нормирования, для того, чтобы средняя интенсивность оставалась неизменной. Полученное значение присваивается

пикселю, которому соответствует центральный элемент матрицы. Пример вычислений можно увидеть на рисунке 2.

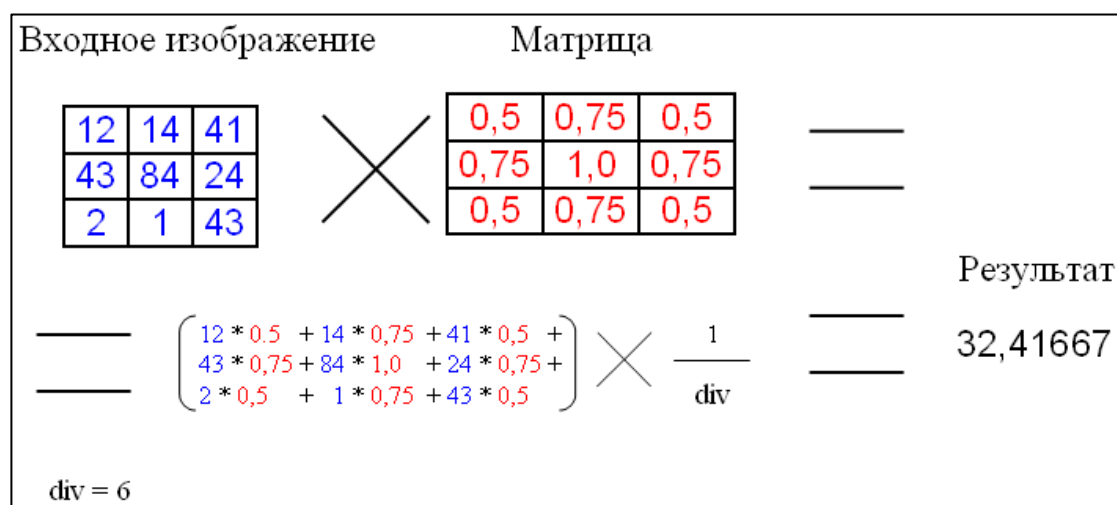


Рисунок 2 — Пример вычислений при фильтрации изображения

Размер квадратной матрицы берется не случайно. Используются нечетные числа, для того чтобы можно было определить однозначно центральный элемент матрицы. В случае, если размер матрицы меньше изображения, то данная фильтрация называется локальной, если же размер матрицы превышает размер изображения – глобальной [7].

Линейные фильтры представляют собой семейство фильтров, имеющих очень простое математическое описание. А также данные фильтры позволяют получить самые разные эффекты. Будем считать, что задано исходное полутоновое изображение  $A$ , и обозначим интенсивности его пикселей  $A(x, y)$ . Линейный фильтр определяется функцией  $F$ , заданной на растре. Данная функция  $F$  является ядром фильтра, а при помощи взвешенного суммирования производится сама фильтрация.

$$B(x, y) = \sum_i \sum_j F(i, j) * A(x + i, y + j), \quad (1.1)$$

где  $B(x, y)$  — изображение, полученное в результате фильтра,

$F(i, j)$  — ядро фильтра,

$A(x + i, y + j)$  — пиксель исходного изображения, на который воздействует фильтр.

В результате фильтрации получаем изображение  $B$ . В указанной выше формуле были опущены пределы суммирования. Обычно  $F(i, j)$  отлично от нуля только в некоторой области  $N$  точки  $(0, 0)$ , за пределами этой области  $N$  значение  $F(i, j)$  равно нулю или настолько близко к нулю, что его значением можно пренебречь. Значение  $B(x, y)$  определяется пикселями исходного изображения  $A$ , которые лежат в области  $N$ . Область  $N$  представляет собой прямоугольную матрицу, у которой длины сторон являются нечетными числами. В случае если пиксель  $(x, y)$  находится на краю, то вычисление невозможно. Подобную проблему возможно решить несколькими способами:

- не проводить фильтрацию крайних пикселей;
- не использовать пиксель в суммировании, распределив его вес  $F(i, j)$  равномерно среди других пикселей области  $N(x, y)$ ;
- доопределить значения пикселей за границей при помощи экстраполяции;
- доопределить значения пикселей за границей при помощи зеркального отражения [8].

Цифровые фильтры по способу выполнения фильтрации могут разделяться на два вида – рекурсивные и нерекурсивные (простые) фильтры. Рассмотренный выше вид фильтрации является нерекурсивным, так как выходные значения определяются только через входные значения фильтра. Рекурсивные фильтры отличаются тем, что выходные значения определяются не только через входные значения фильтра, а также через соответствующие выходные значения.

На выходе самого простого линейного фильтра получается среднее значение окружающих элементов с влиянием маски фильтра. Подобные фильтры также называют сглаживающими или усредняющими. Из-за того, что исходные значения элементов заменяются на усредненные по маске фильтра,

происходит уменьшение резких изменений яркости. В результате решаются следующие задачи:

- подавление шума, который образуется из-за резких перепадов значений яркости;
- подавление деталей, которые не несут смысловой нагрузки, размеры которых малы по сравнению с размерами маски фильтра;
- сглаживание ложных контуров, возникающих из-за оцифровки или преобразований с использованием недостаточного числа уровней яркости[9].

Из минусов работы подобных фильтров, хотелось бы отметить размытие полезных контуров, которые также имеют резкие перепады значений яркостей. На рисунке 3 представлен результат работы данного фильтра.

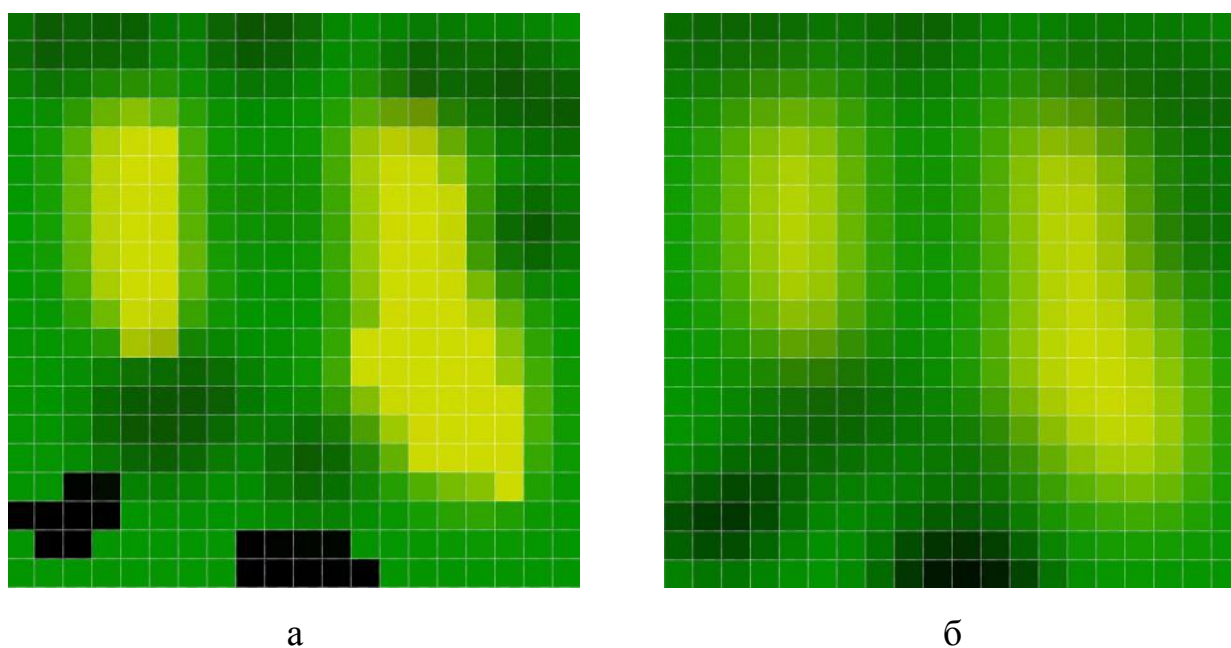


Рисунок 3 — Линейный фильтр: а – оригинал; б – результат работы фильтра

На рисунке 4 изображена матрица, которая охватывает область изображения на оригинальном изображении и полученный в результате работа фильтра пиксель.

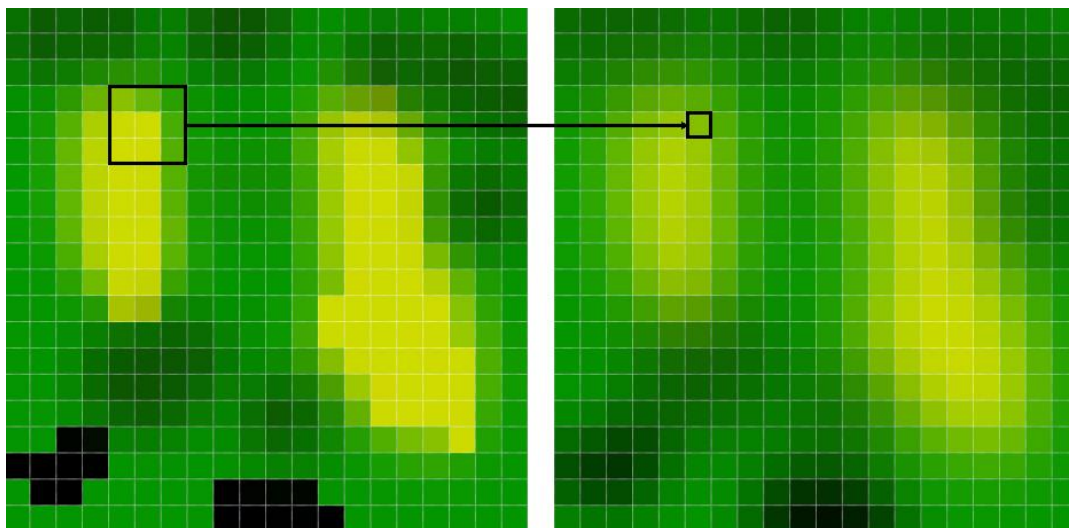


Рисунок 4 — Пример расположения матрицы фильтра (первый случай)

На рисунке 5 изображена та же матрица, смещенная на один пиксель вниз.

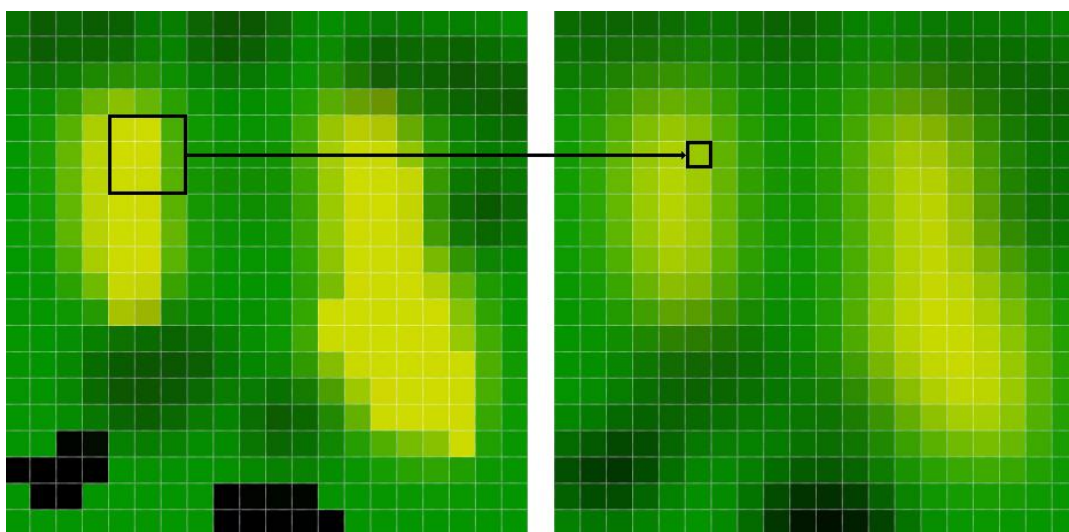


Рисунок 5 — Пример расположения матрицы фильтра (второй случай)

Усредняющий фильтр может быть двух видов — однородный и неоднородный. У однородного фильтра все значения в матрице одинаковые. У неоднородного фильтра каждое значение может быть разным. Чаще всего в неоднородных фильтрах центральное значение имеет наибольший вес и чем дальше от центра, тем значение меньше. Это сделано для уменьшения



расфокусировки изображения при сглаживании. На рисунке 6 изображен пример подобных матриц и их нормировочные коэффициенты [10].

$\frac{1}{9} \times$	<table><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	1	1	1	1	1	1	$\frac{1}{16} \times$	<table><tr><td>1</td><td>2</td><td>1</td></tr><tr><td>2</td><td>4</td><td>2</td></tr><tr><td>1</td><td>2</td><td>1</td></tr></table>	1	2	1	2	4	2	1	2	1
1	1	1																			
1	1	1																			
1	1	1																			
1	2	1																			
2	4	2																			
1	2	1																			
Однородный		Неоднородный																			

Рисунок 6 — Пример матрицы однородного и неоднородного усредняющего фильтра

Пользоваться сглаживающим фильтром нужно осторожно. Он действует на изображение как мутная линза. Изображение становится размытым. Светлые элементы на фоне темных, становятся также темнее. Чем больше матрица фильтрации, тем будет меньше шумов, но при этом будет потеряно больше значащих деталей изображения.

Наиболее эффективным неоднородным фильтром является гауссовский фильтр:

$$F_{gauss}(i, j) = \frac{1}{2\pi\sigma^2} e^{-\frac{i^2+j^2}{2\sigma^2}}, \quad (1.2)$$

где  $F_{gauss}(i, j)$  — ядро фильтра Гаусса,

$\sigma$  — степень размытости.

Гауссовский фильтр имеет ненулевое ядро бесконечного размера. Однако чем дальше от центра, тем оно ближе к нулю, именно поэтому на практике чаще всего используют матрицу небольшого размера, так как большие матрицы окажут не такое значительно влияние на изображение, но ресурсов на обработку потребуется значительно больше.

Гауссовская фильтрация также является сглаживающей. В отличие от прямоугольного фильтра, данная фильтрация работает не с точками, а с симметричным размытым пятном. Влияние пикселей друг на друга в фильтрации Гаусса зависит квадрату расстояния между ними. Чем дальше, тем меньше воздействие. Степень размытия и коэффициент отношения определяются параметром  $\sigma$  [11].

Противоположностью сглаживающих фильтров, которые снижают контрастность и размывают изображение, являются контрастоповышающие фильтры. Сумма значений матрицы равна единице, при этом центральное значение элемента матрицы больше единицы. Например, контрастоповышающий фильтр может иметь матрицу (1.3).

$$M_1^{contr} = \begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}. \quad (1.3)$$

Контраст изображения повышается благодаря тому, что фильтр выделяет разницу между значениями соседних пикселей. Чем больше значение центрального элемента матрицы, тем более сильный будет эффект фильтрации [12].

### 1.3 Оценка качества изображения

Применяя на изображение фильтры, происходит коррекция яркости. Для анализа яркости можно использовать гистограмму распределения яркости. Гистограмма яркости помогает оценить общее качество изображения на основе математической интерпретации тонального диапазона изображения в виде особой диаграммы. Гистограммой цифрового изображения с уровнями яркости в диапазоне  $[0, L-1]$  называется дискретная функция.

Для нормализации гистограмм производят деление каждого значения на общее количество пикселей изображения  $n$ .

Изображения со слабой контрастностью имеют гистограмму яркости, которая сгруппирована в небольшом диапазоне значений. Гистограмма таких изображений может быть смещена в область темных или светлых тонов, или может размещаться и в центре диапазона яркости. Если изображение обладает хорошей контрастностью, то его гистограмма занимает весь диапазон яркостей. Подобные изображения обычно считаются как более качественные. Также могут быть высококонтрастные изображения. У них может также быть задействован весь диапазон яркости, но гистограмма такого изображения может обладать пиками из-за наличия областей изображения светлых или темных тонов [13].

Для примера на рисунке 7 приведено изображение и его гистограмма.

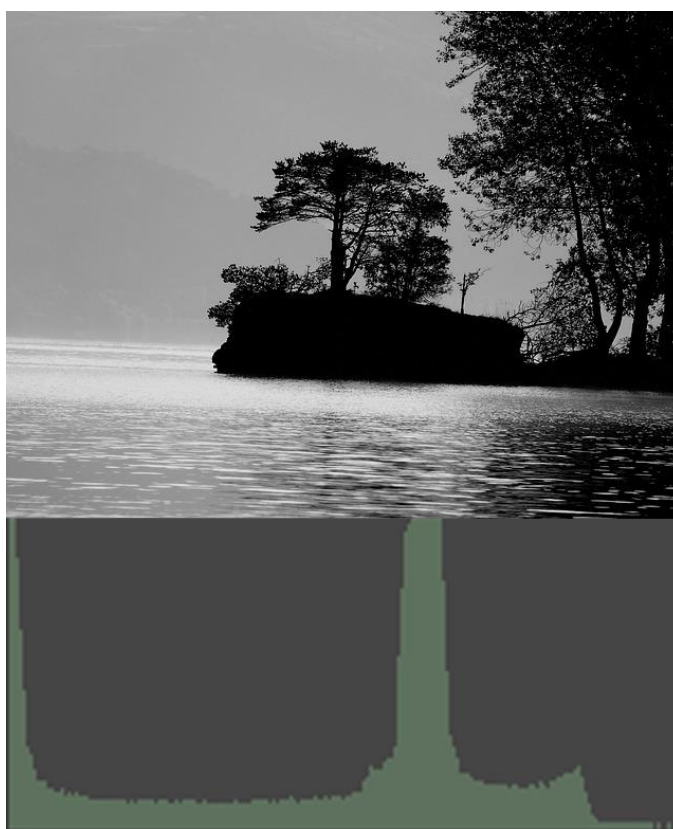


Рисунок 7 — Пример изображения с гистограммой яркости

Левая граница гистограммы соответствует нулевому уровню яркости (черный цвет). Правая граница гистограммы же соответствует максимальному уровню яркости (белый цвет). Уровни между данными границами и создают

изображение. Чем больше на изображении пикселей одной яркости, тем выше уровень на гистограмме. На данном изображении с помощью гистограммы можно увидеть, что имеется два пика. Очень темные цвета и немного светлее среднего цвета.

Идеальной гистограммой считается пологая кривая Гаусса. В ней должно быть минимум самых темных цветов и минимум самых ярких цветов. И чем ближе к средним цветам, тем их уровень должен быть выше. Но на практике такое встречается очень редко и практически невозможно.

В случае если гистограмма изображения является узкой и расположена вблизи центра диапазона яркостей, то можно сделать вывод, что изображение является низкоконтрастным [14].

Для того чтобы увидеть влияние шумов на гистограмму яркости, можно посмотреть на рисунке 8 и 9.

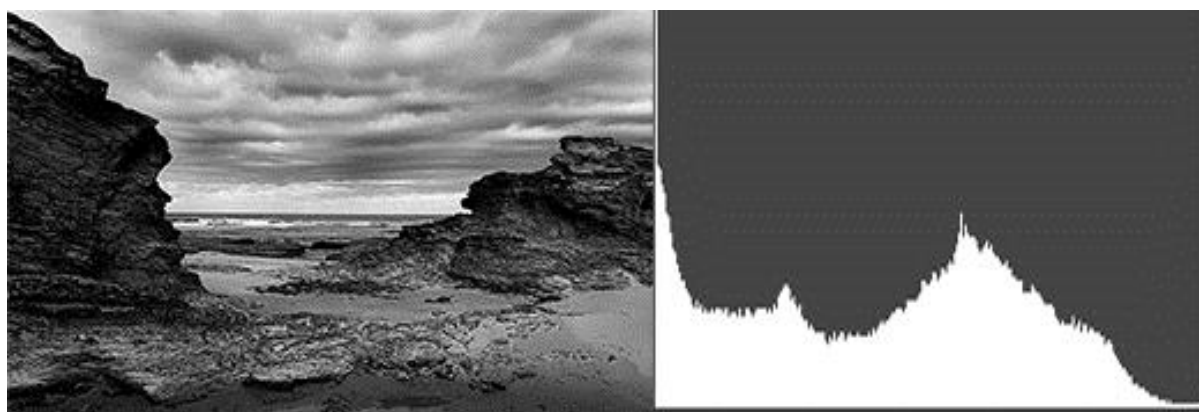


Рисунок 8 — Оригинальное изображение и его гистограмма яркости

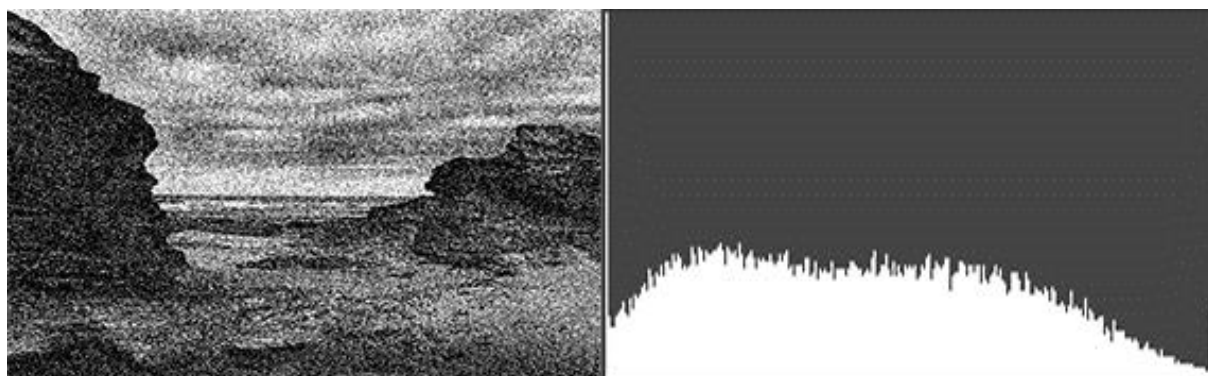


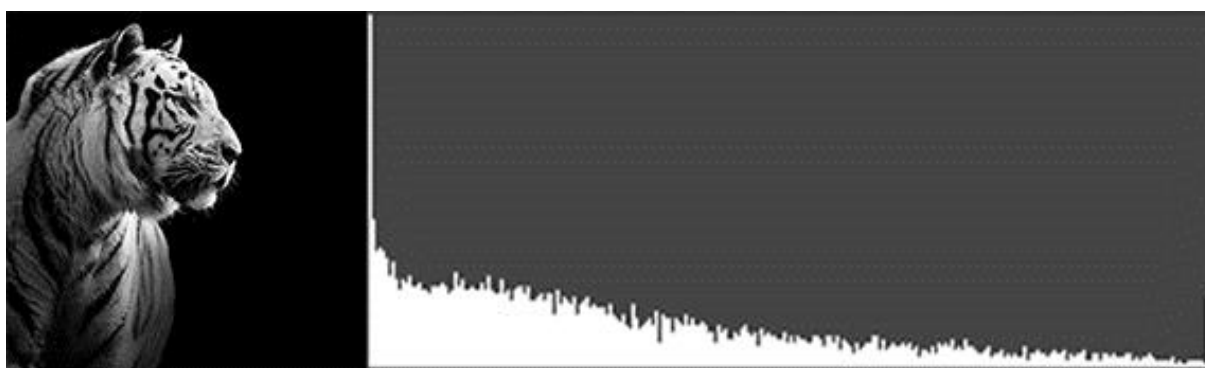
Рисунок 9 — Изображение с шумом и его гистограмма яркости

В оригинальном изображении на гистограмме яркости соседние уровни яркости относительно равны и резкие перепады практически отсутствуют. Если же рассмотреть гистограмму яркости зашумленного изображения, то можно увидеть постоянные резкие перепады

Для демонстрации работы фильтров, необходимо также рассмотреть изменения гистограмм яркости. На рисунке 10 представлено оригинальное изображение и его гистограмма яркости, а далее изображения после применения фильтров – шумоподавление и увеличение резкости краев, и их гистограммы. Отчетливо видно их влияние на гистограмму.

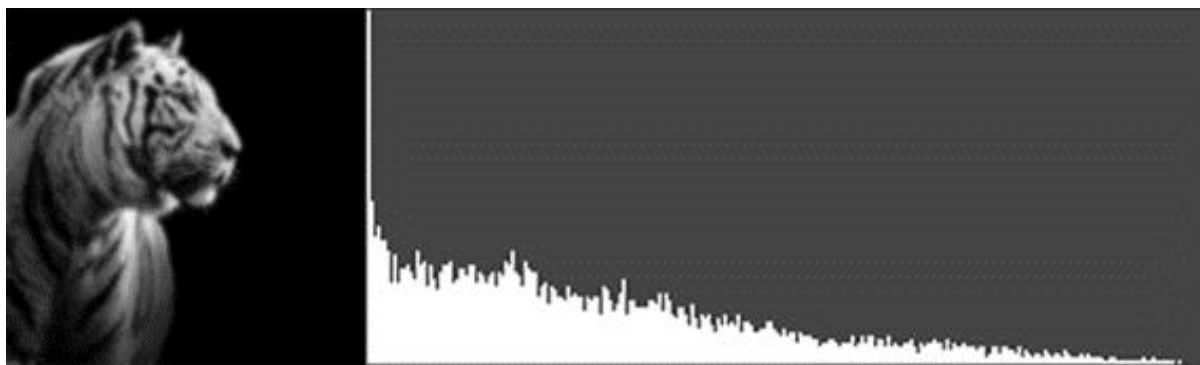
Шумоподавляющий фильтр уменьшает количество сильно выделяющихся уровней, сглаживая гистограмму. Это происходит исходя из алгоритма фильтра, в результате которого происходит усреднение яркости пикселей, а это делает гистограмму плавнее. Фильтр, увеличивающий резкость краев сделал обратную работу, сделав гистограмму более резкой, при этом усилив уровни в области темных и светлых краев.

Вследствие этого, изучив гистограмму исходного изображения и после применения фильтров возможно определить оптимальный алгоритм фильтрации для достижения требуемого качества изображения.

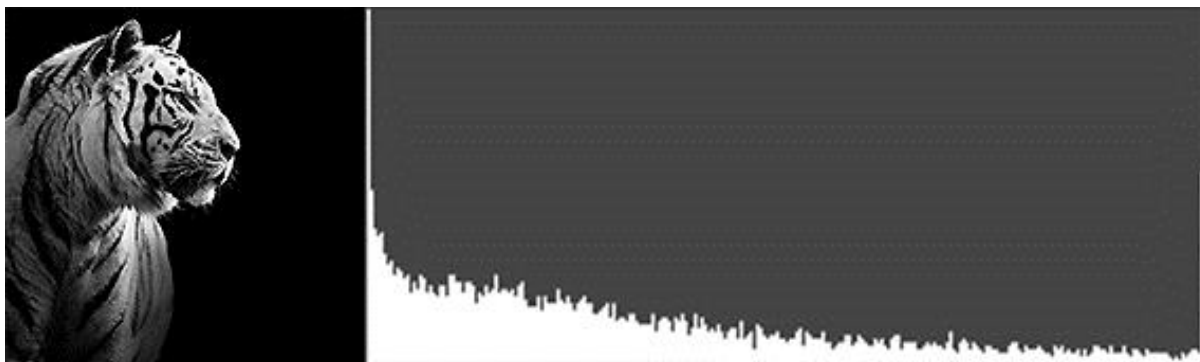


а

Рисунок 10 — Влияние фильтра на гистограмму: а – исходное изображение; б – шумоподавляющий фильтр; в – фильтр, увеличивающий резкость краев, лист 1



б



в

Рисунок 10, лист 2

#### 1.4 Вывод по первой главе

В данной главе рассмотрены причины появления шума на изображениях, а также проанализированы задачи предварительной обработки изображения. Рассмотрены различные виды шумов и фильтры, способные их подавить. Выведено определение гистограммы распределения яркости. Приведены примеры влияния на гистограмму изображения разных фильтров.

По итогу главы сделан вывод, что оценивая качество изображения по его гистограмме распределения яркости и рассмотренных фильтров можно создать оптимальный алгоритм фильтрации изображений. Помимо тех фильтров, которые были рассмотрены, существует большое количество других. Именно поэтому очень важно определиться с оптимальным фильтром.

## **Глава 2. Практическая часть**

### **2.1 Диаграмма вариантов использования**

Для проектирования функциональной модели модуля фильтрации спутниковых изображений необходимо выполнить построение диаграммы вариантов использования (Use Case Diagrams). Данная диаграмма необходима для описания функционального назначения системы.

Диаграмма вариантов использования демонстрирует действующих лиц (актеров или исполнителей), прецеденты и отношения между ними:

Актер — личность, организация или система, взаимодействующая с информационной системой. Существует два вида актеров: внешний актер, который использует или используется системой, порождая прецеденты деятельности и внутренний актер, который обеспечивает реализацию прецедентов деятельности внутри системы [15].

Прецедент — законченный алгоритм, который инициирован внешним объектом (личность или система), который взаимодействует с информационной системой и получает в результате определенное сообщение от информационной системы.

Актерами разрабатываемого модуля являются: Оператор, Сервис мониторинга неоднородностей. Оператор выбирает необходимую функцию, а сервис мониторинга неоднородностей выполняет ее. Обобщенная диаграмма использования представлена на рисунке 11.

Варианты использования «Оператор»:

- применение фильтра Гаусса: применение алгоритма фильтра Гаусса на указанное изображение с определенным значением фильтра;
- применение медианного фильтра: применение алгоритма медианного фильтра на указанное изображение с определенным значением фильтра;
- векторизация: преобразование растрового изображения в векторное.

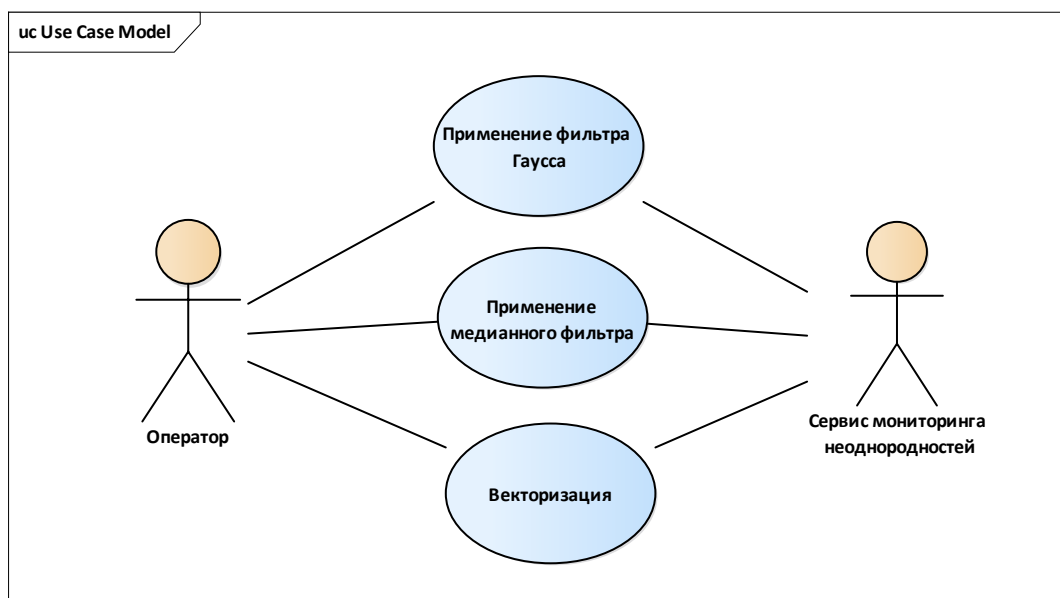


Рисунок 11 — Обобщенная диаграмма использования

## 2.2 Диаграмма деятельности

При моделировании поведения проектируемой системы часто возникает необходимость в детализации алгоритмической и логической реализации выполняемых системой операций. Для этого используются диаграммы деятельности [16].

Диаграммы деятельности можно использовать для моделирования динамических аспектов поведения системы. Как правило, они применяются, чтобы промоделировать последовательные (а иногда и параллельные) шаги вычислительного процесса.

Диаграмма деятельности (activity diagram) напоминает блок-схему описания алгоритма. В ней точками принятия решений и переходов описывается последовательность шагов (названных с достаточной точностью видами деятельности). Такая диаграмма достаточно удобна для отображения бизнес-процессов или операций. Поэтому диаграммы деятельности являются неотъемлемой частью системного анализа.

Каждый вид деятельности (функция) представлен прямоугольником с закругленными углами. После завершения одного вида деятельности переход к



другому осуществляется автоматически. Переход от одного вида деятельности к другому обозначен стрелкой. Также на диаграмме имеется начальная точка, изображенная в виде закрашенного кружка и конечная точка в виде закрашенного кружка с обводом. Данные точки являются точками входа и выхода в диаграмму деятельности

Как правило, в работе каждой системы возникает момент, когда система должна осуществить выбор между двумя путями развития. Один набор условий выводит на один путь, следующий — на другой путь, причем эти пути исключают друг друга. Точка принятия решения изображается в виде ромба, из которого выходят возможные пути.

Далее будут рассмотрены следующие прецеденты:

- применение фильтра Гаусса;
- применение медианного фильтра;
- векторизация.

#### **Прецедент 1: применение фильтра Гаусса**

**Краткое описание.** Использование алгоритма фильтра Гаусса на выбранное изображение с определенными параметрами и сохранение в указанном месте.

Действующее лицо прецедента — Оператор системы.

Базовый поток — Применение фильтра Гаусса:

- оператор выбирает из предложенных функций применение фильтра Гаусса;
- система мониторинга неоднородностей запрашивает входные данные, необходимые для работы функции;
- оператор выбирает изображение, на которое нужно применить фильтр Гаусса;
- оператор вводит числовое значение, определяющее силу фильтра;
- оператор вводит путь, куда будет произведено сохранение результата работы фильтра;
- осуществляется применение алгоритма фильтра Гаусса;

– производится сохранение результата работы функции.

**Постусловие.** При успешном окончании прецедента обработанное изображение находится в указанном оператором месте.

На входе алгоритма фильтра Гаусса от Оператора системы требуется ввести

Прецедент в виде диаграммы деятельности представлен на рисунке 12.

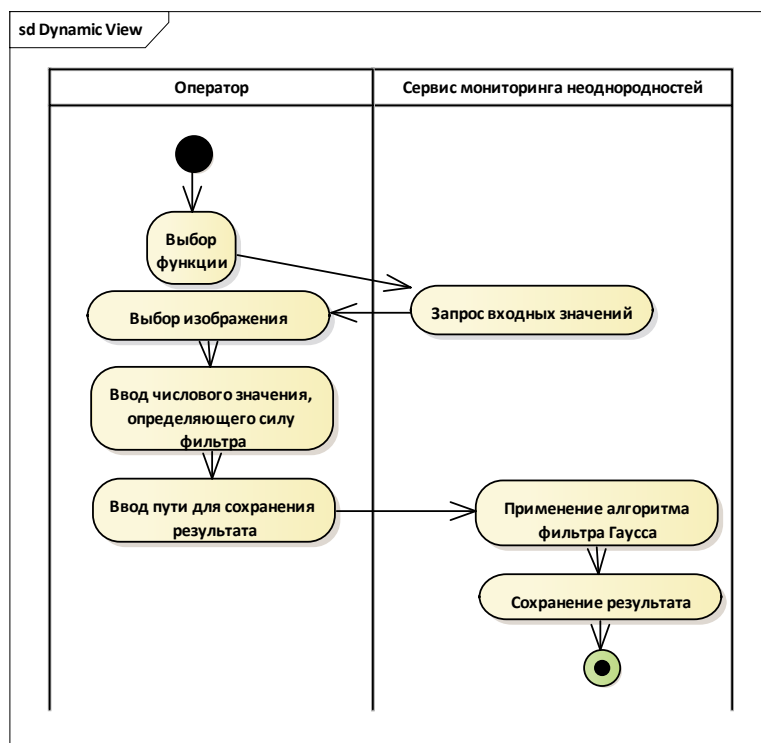


Рисунок 12 — Прецедент «Применение фильтра Гаусса»

## Прецедент 2: применение медианного фильтра

**Краткое описание.** Использование алгоритма медианного фильтра на выбранное изображение с определенными параметрами и сохранение в указанном месте.

Действующее лицо прецедента — Оператор системы.

Базовый поток — Применение медианного фильтра:

– оператор выбирает из предложенных функций применение медианного фильтра;

- система мониторинга неоднородностей запрашивает входные данные, необходимые для работы функции;
- оператор выбирает изображение, на которое нужно применить медианный фильтр;
- оператор вводит числовое значение, определяющее силу фильтра;
- оператор вводит путь, куда будет произведено сохранение результата работы фильтра;
- осуществляется применение алгоритма медианного фильтра;
- производится сохранение результата работы функции.

**Постусловие.** При успешном окончании прецедента обработанное изображение находится в указанном оператором месте.

Прецедент в виде диаграммы деятельности представлен на рисунке 13.

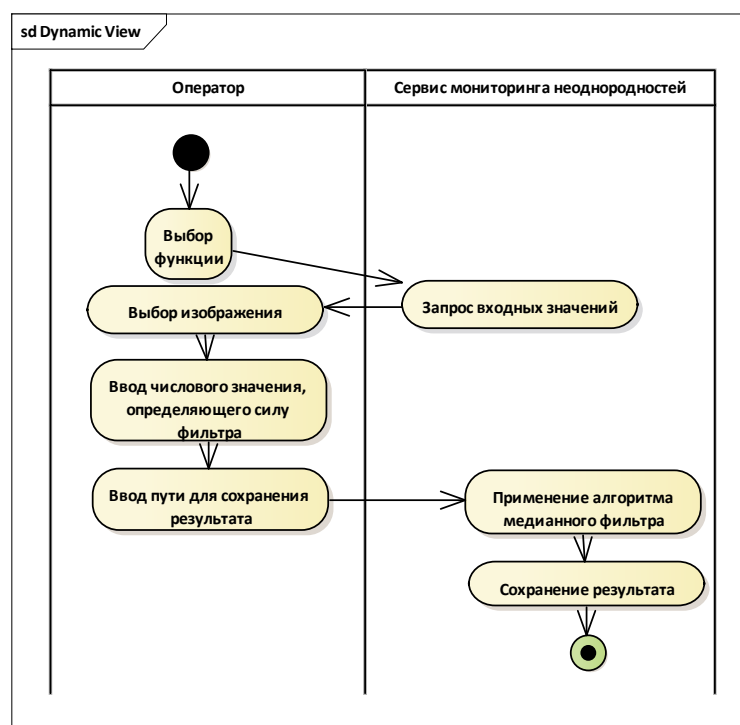


Рисунок 13 — Прецедент «Применение медианного фильтра»

### Прецедент 3: векторизация

**Краткое описание.** Использование векторизации выбранного растрового изображения и сохранение результата в указанном месте.

Действующее лицо прецедента — Оператор системы.

Базовый поток — Векторизация:

- оператор выбирает из предложенных функций векторизацию;
- система мониторинга неоднородностей запрашивает входные данные, необходимые для работы функции;
- оператор выбирает изображение, на которое нужно применить векторизацию;
- оператор вводит путь, куда будет произведено сохранение результата работы фильтра;
- осуществляется применение алгоритма векторизации;
- производится сохранение результата работы функции.

**Постусловие.** При успешном окончании прецедента векторное изображение находится в указанном оператором месте.

Прецедент в виде диаграммы деятельности представлен на рисунке 14.

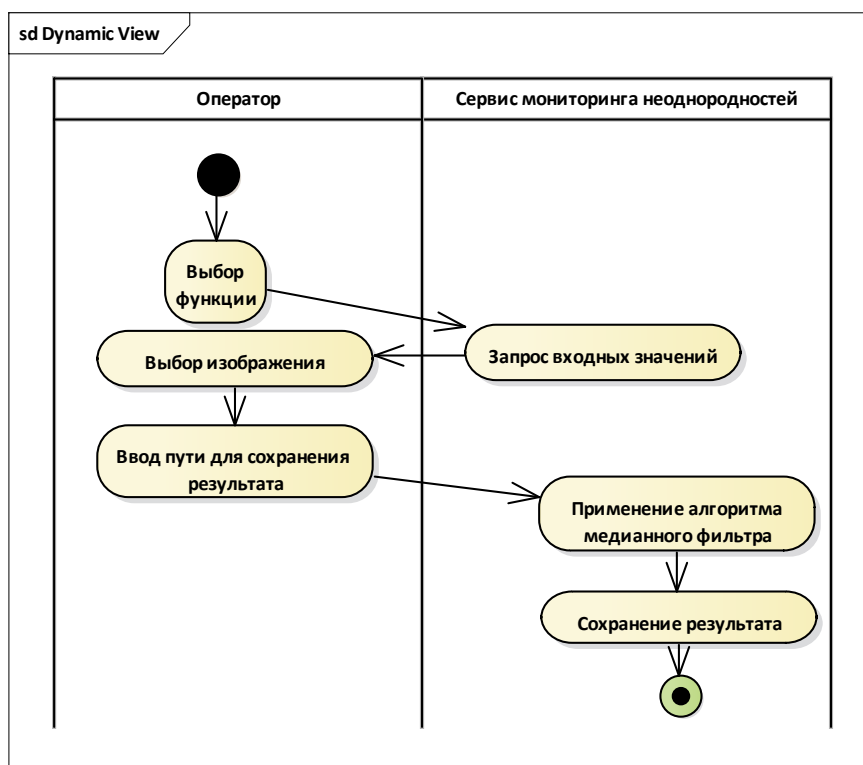


Рисунок 14 — Прецедент «Векторизация»

## 2.3 Диаграмма развертывания

Для моделирования элементов и компонентов программы, существующих на этапе ее исполнения, используются диаграммы развертывания (deployment diagram). С их помощью перед разработкой можно определиться с необходимыми библиотеками, которые будут использоваться приложением, интерпретатором, обрабатывающим и выполняющим программный код и со средой выполнения.

Каждый узел диаграммы развертывания представлен в виде параллелепипеда. Связи между узлами диаграммы изображены в виде линий.

В разрабатываемом модуле необходимо использование библиотеки GDAL и библиотеки OGR. Библиотека GDAL работает с растровыми изображениями. С помощью него происходит открытие изображения, получения информации, извлечение растра или определенного канала и последующее сохранение изменений. Библиотека OGR выполняет аналогичные функции, но работает с векторными изображениями.

Схема взаимодействия компонентов модуля в виде диаграммы компонентов представлена на рисунке 15.

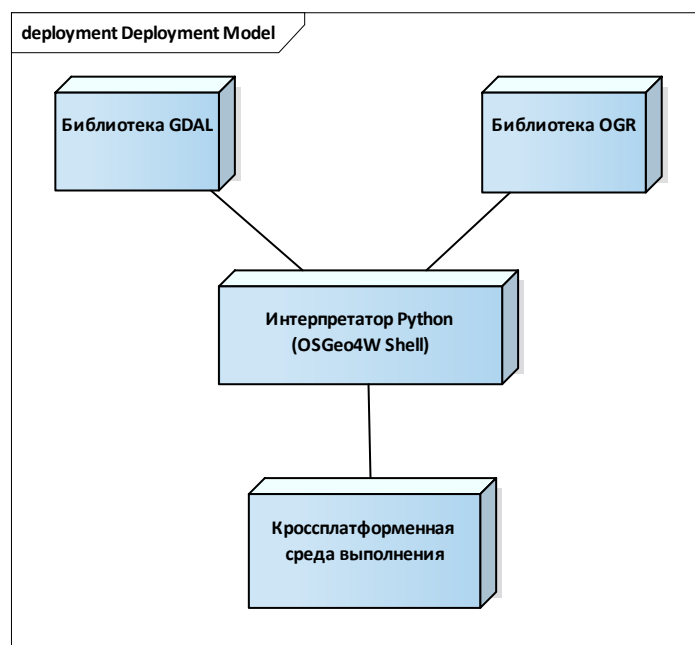


Рисунок 15 — Схема взаимодействия компонентов модуля

## **2.4 Разработка модуля QGIS**

Модули - это отличный способ расширить функциональность QGIS. Для создания модулей используется язык программирования Python, который позволяет создавать модули самой разной сложности.

### **2.4.1. Установка необходимых модулей QGIS**

Для разработки модуля QGIS необходимы два модуля: Plugin Builder и Plugin Reloader.

Plugin Builder используется для создания необходимых файлов модуля и шаблонного кода, который обеспечивает основные функции модуля (открытие, закрытие и т.п.).

Plugin Reloader используется для обновления модуля после внесения в его код правок.

### **2.4.2. Разработка интерфейса модуля**

Для разработки интерфейса модуля используется программа Qt Creator. С ее помощью возможно открыть файл с расширением .ui, находящийся в папке разрабатываемого модуля, созданной модулем Plugin Builder.

Qt Creator является кроссплатформенной интегрированной средой разработки. А также помимо разработки интерфейса позволяет писать и компилировать программный код на таких языках программирования как С и С++. Разработка интерфейса осуществляется с помощью декларативного языка программирования QML, основанном на JavaScript.

Qt Creator позволяет разместить практически все основные элементы интерфейса и настроить их свойства. Интерфейс программы можно наблюдать на рисунке 16.

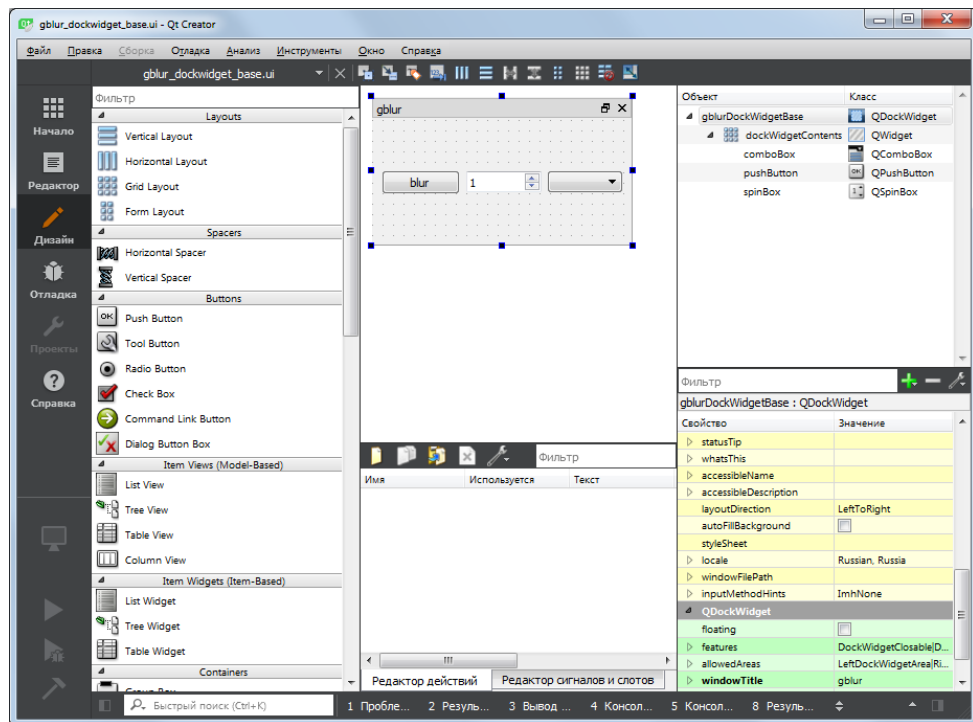


Рисунок 16 — Qt Creator

### 2.4.3. Разработка программного кода

Для написания программного кода используется язык программирования Python. В качестве среды написания можно использовать любой текстовый редактор. Например, Notepad++. В папке разрабатываемого модуля, созданной модулем Plugin Builder, имеется одноименный файл с расширением \*.ру, который содержит программный код модуля. Интерфейс программы представлен на рисунке 17.

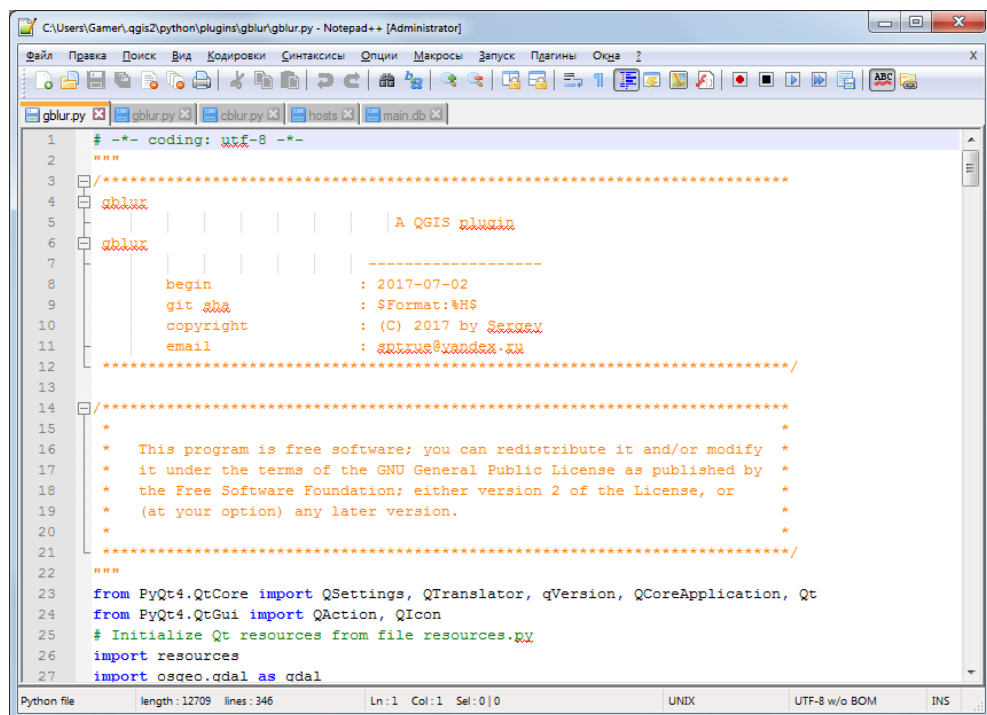


Рисунок 17 — Notepad++

## 2.5 Программная реализация фильтра Гаусса

Фильтр Гаусса является матричным фильтром. Принцип его работы был рассмотрен в первой главе. Значения матрицы рассчитываются следующим образом:

$$window(i) = \begin{cases} e^0 = 1, & \text{если } i = 0; \\ e^{-i^2/(2\sigma^2)}, & \text{если } i \in 1..N; \\ window(-i), & \text{для отрицательных.} \end{cases} \quad (1.4)$$

Разработана блок-схема реализации алгоритма фильтра Гаусса. Построенная блок-схема представлена на рисунке 18.



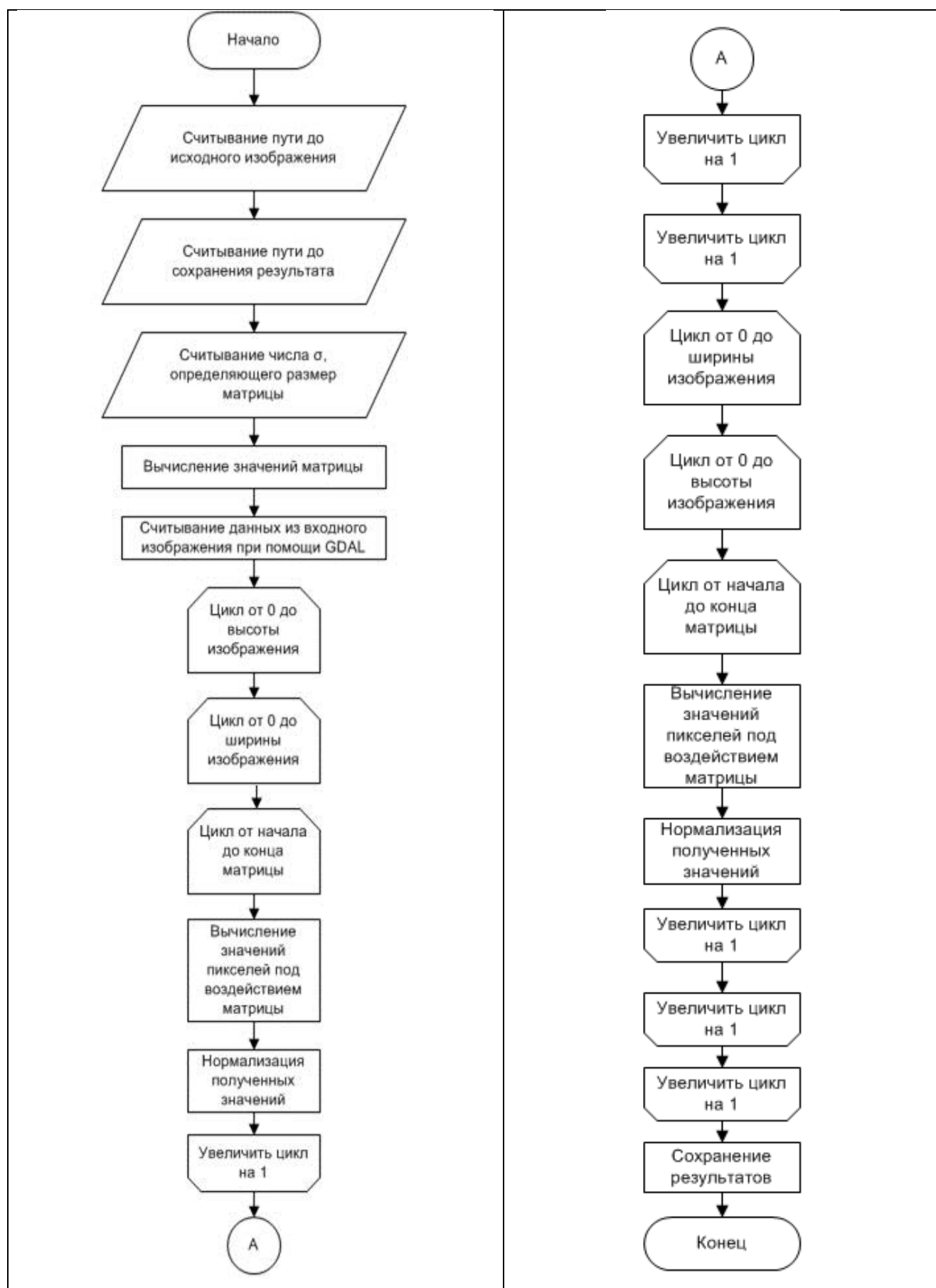


Рисунок 18 — Блок-схема реализации алгоритма фильтра Гаусса

Программный код представлен в приложении А. Интерфейс разработанного модуля, можно наблюдать на рисунке 19.

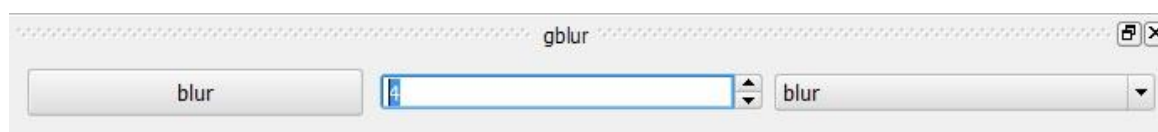


Рисунок 19 — Интерфейс разработанного модуля

## 2.6 Программная реализация медианного фильтра

Медианный фильтр также является матричным фильтром. Пиксели, попадающие в матрицу, сортируются, и определяется среднее значение, которое является выходным для пикселя, на который применялся фильтр.

Разработана блок-схема реализации медианного фильтра. Построенная блок-схема представлена на рисунке 20.

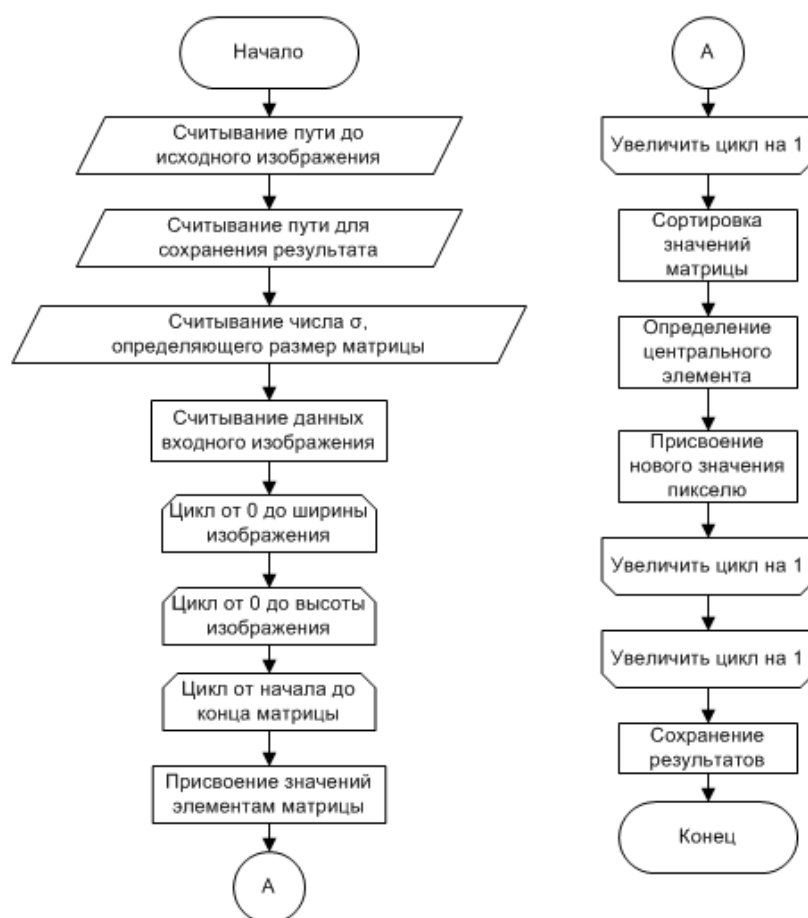


Рисунок 20 — Блок-схема реализации медианного фильтра

Программный код представлен в приложении Б. Интерфейс разработанного модуля идентичный модулю, реализующего алгоритм фильтра Гаусса, его можно наблюдать на рисунке 21.

Имеется поле для ввода числового значения, определяющего силу фильтра, выбор слоя из слоев проекта и кнопка, которая запускает работу модуля, реализующего алгоритм медианного фильтра.

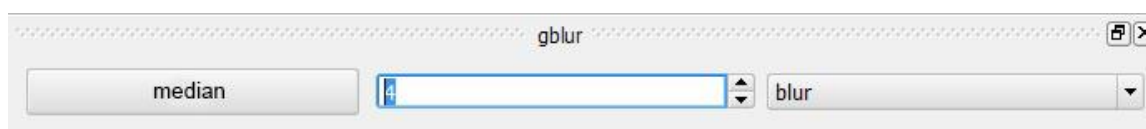
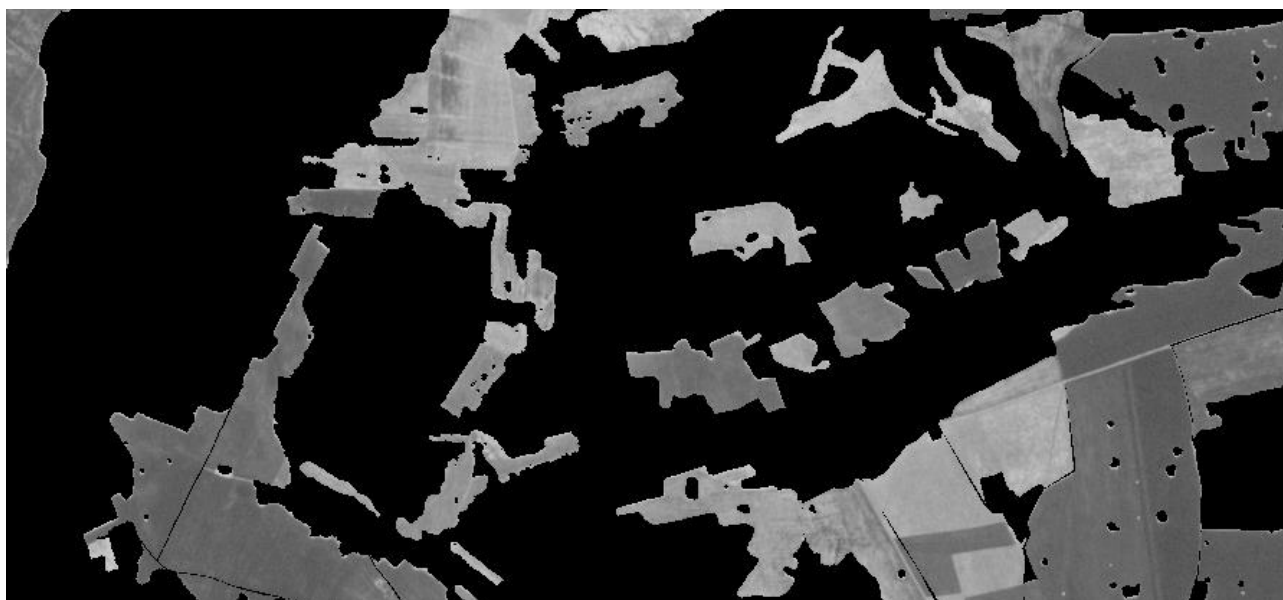


Рисунок 21 — Интерфейс разработанного модуля

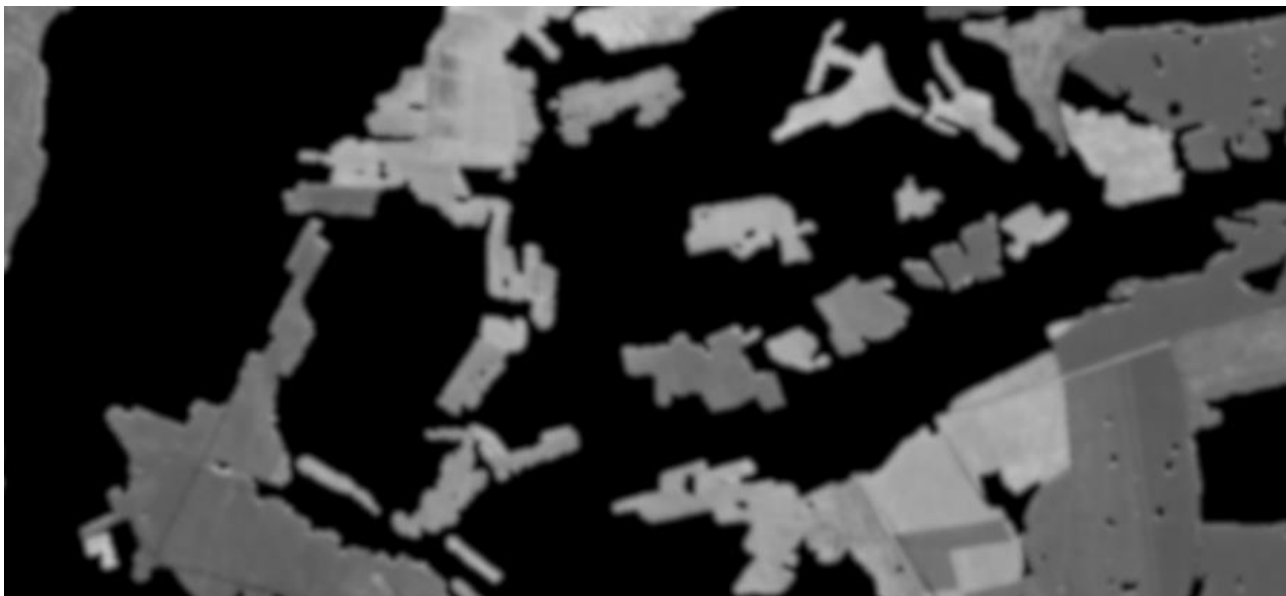
## 2.7 Результаты работы модулей

В качестве примера изначально выбран фрагмент несегментированного спутникового изображения и применены к нему фильтр Гаусса и медианный фильтр. Результаты их работы представлены на рисунке 22.

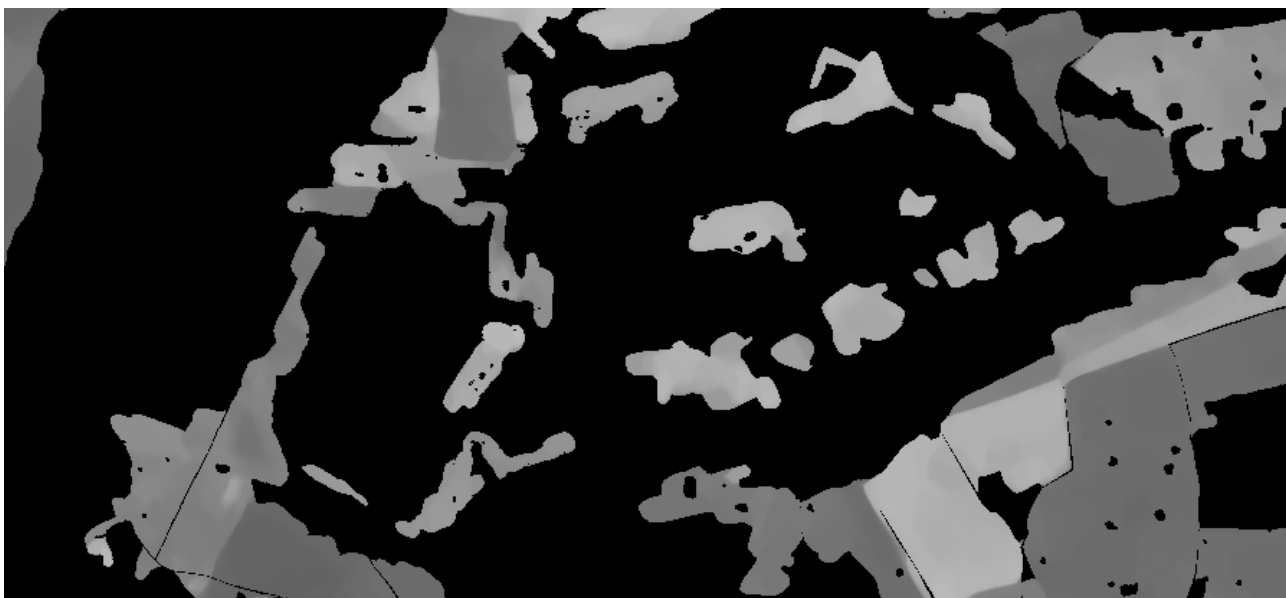


а

Рисунок 22 — Влияние фильтра на изображение: а – исходное изображение; б – фильтр Гаусса; в – медианный фильтр, лист 1



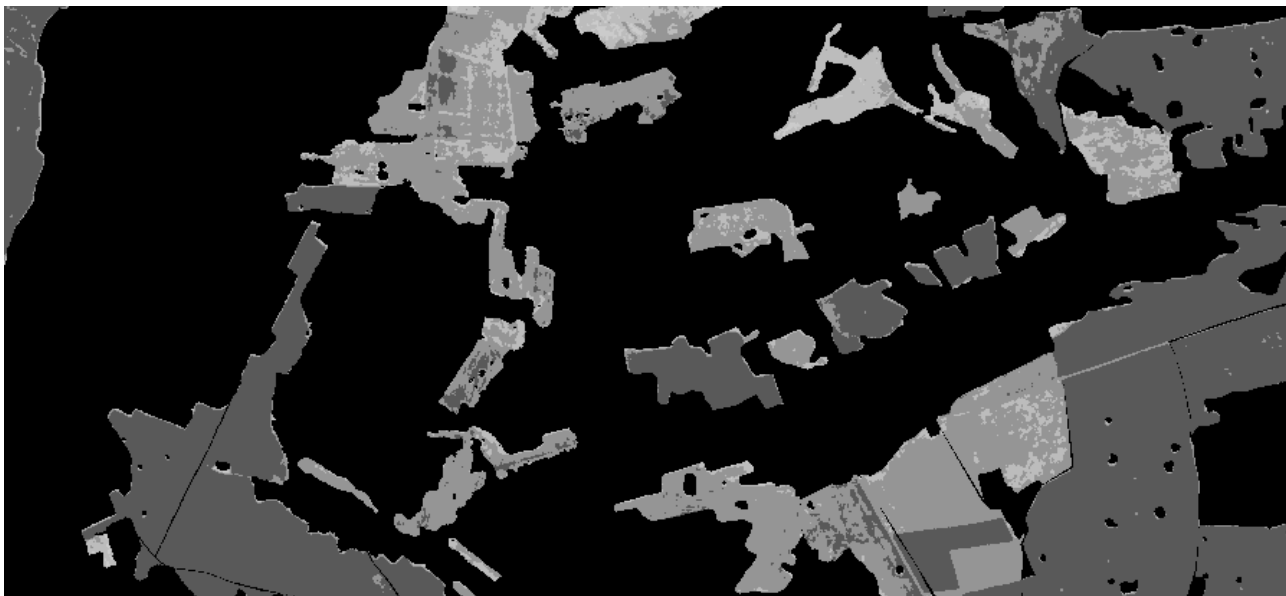
б



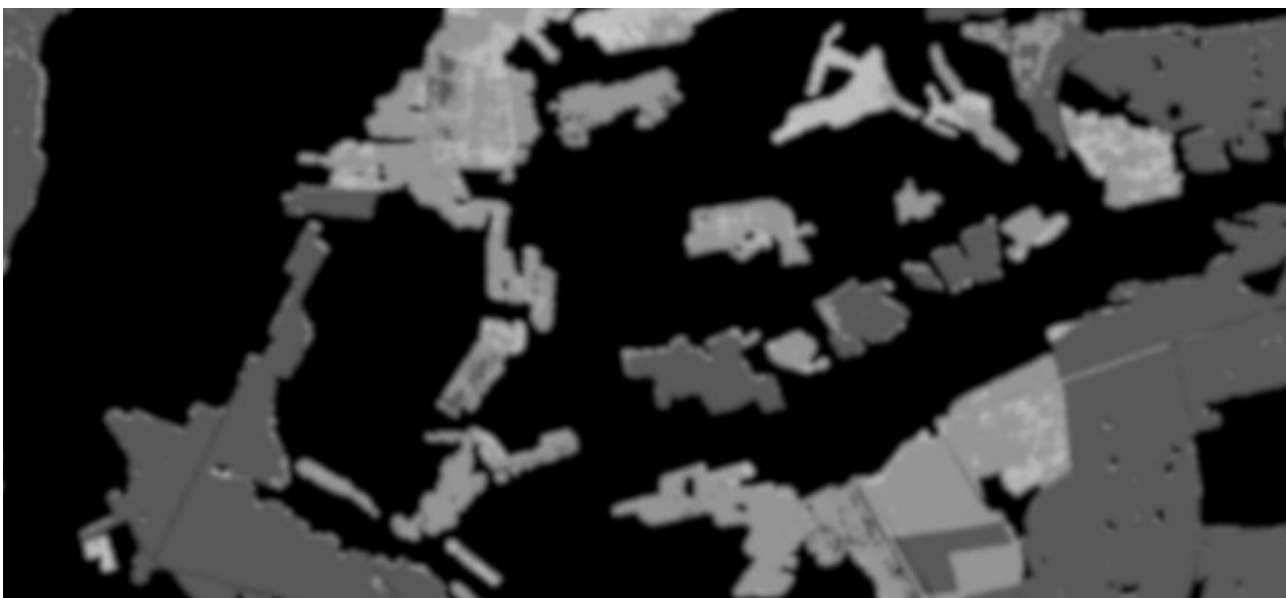
в

Рисунок 22, лист 2

Также применены фильтры к сегментированному фрагменту спутникового изображения. Результаты их работы можно видеть на рисунке 23.

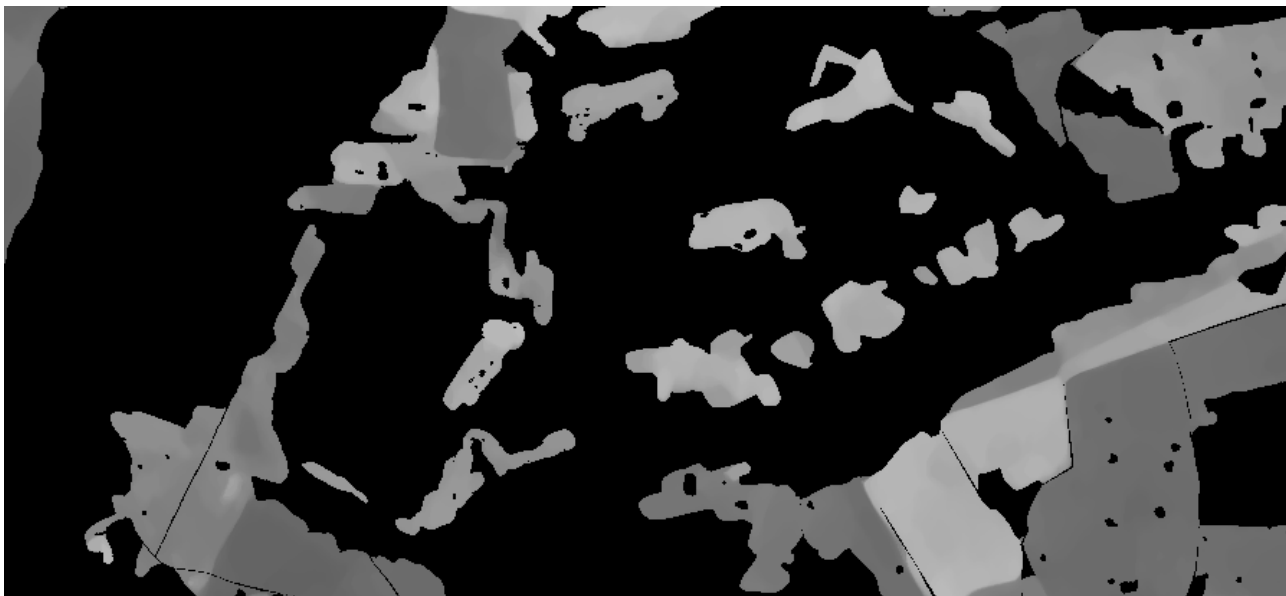


а



б

Рисунок 23 — Влияние фильтра на изображение: а – исходное изображение; б – фильтр Гаусса; в – медианный фильтр, лист 1



В

Рисунок 23, лист 2

## ЗАКЛЮЧЕНИЕ

Использование спутниковых изображений, их сегментация, классификация снимков, использование данных дистанционного зондирования Земли в целом находят большое применение в современных информационных системах. В бакалаврской работе рассматривалась фильтрация спутниковых изображений для их дальнейшей сегментации.

Для проектирования функциональной модели модуля фильтрации спутниковых изображений была построена диаграмма вариантов использования. Для детализации алгоритмической и логической реализации выполняемых системой операций были реализованы диаграммы деятельности.

В результате бакалаврской работы выполнены все поставленные задачи:

- 1) Произведен обзор методов и алгоритмов фильтрации изображений. Рассмотрены виды фильтров, их достоинства и недостатки, а также определен способ оценки качества их работы
- 2) Разработан программный модуль реализующий алгоритм фильтра Гаусса. Данный модуль принимает на вход исходное изображение и числовое значение, определяющее силу фильтра, и на выход выдает отфильтрованное изображение
- 3) Разработан программный модуль реализующий алгоритм медианного фильтра. Данный модуль принимает на вход исходное изображение и выдает отфильтрованное изображение.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Кашкин, В. Б. Цифровая обработка аэрокосмических изображений: учебное пособие / В. Б. Кашкин, А. И. Сухинин ; Сиб. федер. ун-т, Ин-т космич. и информ. технологий. - Красноярск : ИПК СФУ, 2008. - 288 с.
2. Хуанг, Т. С. Обработка изображений и цифровая фильтрация : перевод с английского / под ред. Т. С. Хуанг. - Москва: Мир, 1979. - 318 с.
3. Емельянов, С. Г. Обработка цифровых аэрокосмических изображений для геоинформационных систем : монография / С. Г. Емельянов - Старый Оскол : ТНТ, 2011. - 175 с.
4. Потапов, А. А. Новейшие методы обработки изображений : монография / А. А. Потапов, Ю. В. Гуляев; под ред. А. А. Потапов. - Москва : ФИЗМАТЛИТ, 2008. - 496 с.
5. Басараб, М. А. Цифровая обработка сигналов и изображений в радиофизических приложениях : монография / М. А. Басараб ; ред. В. Ф. Кравченко. - Москва : ФИЗМАТЛИТ, 2007. - 544 с.
6. Гонсалес, Р. Цифровая обработка изображений : монография : пер. с англ. / Р. Гонсалес, Р. Вудс. - Москва : Техносфера, 2006. - 1070 с.
7. Грузман, И. С. Цифровая обработка изображений в информационных системах : учебник / И. С. Грузман, В. С. Киричук, В. П. Косых; Новосиб. гос. техн. ун-т. - Новосибирск : Изд-во НГТУ, 2002. - 351 с.
8. Кашкин, В. Б. Дистанционное зондирование Земли из космоса. Цифровая обработка изображений : учеб. пособие / В. Б. Кашкин, А. И. Сухинин. - Москва : Логос, 2001. - 263 с.
9. Гашников, М. В. Методы компьютерной обработки изображений : учебное пособие для студентов, обучающихся по специальности "Прикладная математика" / М. В. Гашников, Н. И. Глумов; под ред. В. А. Сойфер. - Москва : Физматлит, 2001. - 781 с.



10. Очин, Е. Ф. Вычислительные системы обработки изображений / Е. Ф. Очин ; под ред. С. П. Левкович. - Ленинград : Энергоатомиздат, Ленингр. отделение, 1989. – 136 с.
11. Хуанг, Т.С. Быстрые алгоритмы в цифровой обработке изображений : преобразования и медианные фильтры : пер. с англ / Т.С. Хуанг, Дж.-О. Эклунд, Г.Дж. Нуссбаумер и др.; ред. Т.С. Хуанг, пер. Л. П. Ярославский. - Москва : Радио и связь, 1984. - 221 с.
12. Анисимов, Б. В. Распознавание и цифровая обработка изображений : Учеб. пособие для вузов по спец. "ЭВМ" и "АСУ" / Б. В. Анисимов, В. Д. Курганов, В. К. Злобин. - Москва : Высшая школа, 1983. - 295 с.
13. Прэтт, У. К. Цифровая обработка изображений : пер. с англ.; монография / У. К. Прэтт. - Москва : Мир , 1982 – 781 стр.
14. Ярославский, Л. П. Введение в цифровую обработку изображений : научное издание / Л. П. Ярославский. - Москва : Советское радио, 1979. - 312 с.
15. Леоненков, А.В. Самоучитель UML : Самоучитель. 2-е изд., перераб. И доп. / А.В. Леоненков. — СПб : БХВ-Петербург, 2007 – 105 с
16. Фаулер, М. UML. Основы. Краткое руководство по унифицированному языку моделирования. Третье издание: книга / Мартин Фаулер. – СПб: Символ-Плюс, 2004. – 139 с.

## ПРИЛОЖЕНИЕ А

### Программный код фильтра Гаусса

```
import sys
import osgeo.gdal as gdal
from math import exp
import numpy

if __name__ == "__main__":
    sigma = int(sys.argv[3])
    s2=float(2*sigma*sigma)
    n = 3* sigma
    window = [0.0] * (n*2+1)
    window[0] = 1
    for i in range (1,n+1):
        window[i]=exp((-i*i)/s2)
    for i in range (1,n+1):
        window[-i]=exp((-i*i)/s2)
    sum=0.0
    pixr=0.0
    l=0
    outPath=sys.argv[2]
    driver = gdal.GetDriverByName( "GTiff" )
    metadata = driver.GetMetadata()
    bands=1
    dt=gdal.GDT_Float32
    outData=None
    gdalData = gdal.Open(sys.argv[1])
    x=gdalData.RasterXSize
    y=gdalData.RasterYSize
    tmpr= gdalData.GetRasterBand(1).ReadAsArray()
    tmprx = numpy.zeros((y+x))
    outData = driver.Create(outPath, x, y, bands, dt)
    for j in range (0,y):
        for i in range (0,x):
            sum=0.0
            pixr=0.0
```

```

        for k in range (-n,n):
            l=i+k;
            if (l>=0) and (l<x):
                pixr=pixr+tmpr[j][l]*window[k]
                sum=sum+window[k]
        pixr=pixr/sum
        tmprx[i]=pixr
    for i in range (0,x):
        tmpr[j][i]=tmprx[i]
for i in range (0,x):
    for j in range (0,y):
        sum=0.0
        pixr=0.0
        for k in range (-n,n):
            l=j+k
            if (l>=0) and (l<y):
                pixr=pixr+tmpr[l][i]*window[k]
                sum=sum+window[k]
        pixr=pixr/sum
        tmprx[j]=pixr
    for j in range (0,y):
        tmpr[j][i]=tmprx[j]
print('Blured in '+sys.argv[2])
outData.GetRasterBand(1).WriteArray(tmpr)
outData.GetRasterBand(1).FlushCache()

```

## ПРИЛОЖЕНИЕ Б

### Программный код медианного фильтра

```
from osgeo import gdal

import os
import sys
import numpy

try:
    from osgeo import ogr
except ImportError:
    import ogr

currentPath = os.path.dirname(sys.argv[0])

def bandAsArray(path, bandNum):
    gdalData = gdal.Open(path)
    gdalBand = gdalData.GetRasterBand(bandNum)
    array = gdalBand.ReadAsArray().astype(numpy.float32)
    gdalBand = None
    gdalData = None
    return array

def saveRaster(outPath, etalonPath, array):
    gdalData = gdal.Open(etalonPath)
    projection = gdalData.GetProjection()
    transform = gdalData.GetGeoTransform()
    xsize = gdalData.RasterXSize
    ysize = gdalData.RasterYSize
    gdalData = None

    format = "GTiff"
    driver = gdal.GetDriverByName(format)
    metadata = driver.GetMetadata()
    if metadata.has_key(gdal.DCAP_CREATE) and
metadata[gdal.DCAP_CREATE] == "YES":
```

```

        outRaster = driver.Create(outPath, xsize, ysize, 1,
gdal.GDT_Float32)
        outRaster.SetProjection(projection)
        outRaster.SetGeoTransform(transform)
        outRaster.GetRasterBand(1).WriteArray(array)
        outRaster = None
    else:
        print("Driver %s does not support Create() method." %
format)
        return False

def MedianFilter2D(image, frame):
    raster = bandAsArray(image, 1)
    gdalData = gdal.Open(image)
    xsize = gdalData.RasterXSize
    ysize = gdalData.RasterYSize
    gdalData = None
    x = frame // 2
    y = frame // 2
    xs = xsize - x - 1
    ys = ysize - y - 1

    for col in range(x, xs+1):
        for row in range(y, ys+1):
            if raster[row, col] != 0:
                frameArray = []
                for rx in range(col-x, col+x+1):
                    for ry in range(row-y, row+y+1):
                        frameArray.append(raster[ry, rx])

                frameArray.sort()
                mediana = frame ** 2 // 2
                raster[row, col] = frameArray[mediana]

    saveRaster(currentPath + "/NDVI_MF2D_5x5.tif", image,
raster)
    MedianFilter2D(currentPath + "/NDVI.tif", 9)

```

## Плакаты презентации

## Бакалаврская работа

Выпускник С. В. Падалко

Красноярск 2018

Рисунок А.1 — Слайд презентации № 1

## Цель и задачи

- Цель: разработка программного модуля, реализующего фильтр Гаусса и медианный фильтр для применения их к спутниковым изображениям.
- Задачи:
  1. обзор методов и алгоритмов фильтрации изображений;
  2. разработка программного модуля, реализующего алгоритм фильтра Гаусса;
  3. разработка программного модуля, реализующего алгоритм медианного фильтра.

2

Рисунок А.2 — Слайд презентации № 2

## Актуальность

Спутниковые изображения в исходном состоянии использовать неэффективно, так как присутствует большое количество лишних данных, таких как незначительные участки, которые можно опустить при анализе. Для этого используется фильтрация. Она позволяет объединить близлежащие области в одну и подавить маленькие области, которые находятся вдали от больших.

3

Рисунок А.3 — Слайд презентации № 3

## Диаграмма вариантов использования



4

Рисунок А.4 — Слайд презентации № 4



Рисунок А.5 — Слайд презентации № 5

## Инструменты разработки



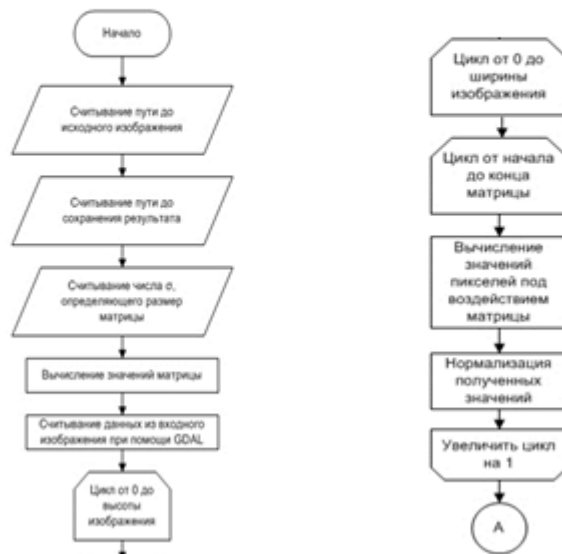

**GDAL** - библиотека для работы с географическими форматами данных. GDAL представляет собой набор утилит для обработки растровых данных.

**OSGeo4W** – интерпретатор Python

Рисунок А.6 — Слайд презентации № 6



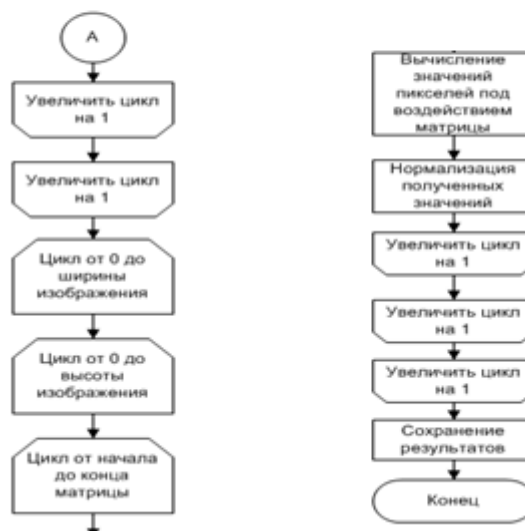
## Блок-схема реализации алгоритма фильтра Гаусса



7

Рисунок А.7 — Слайд презентации № 7

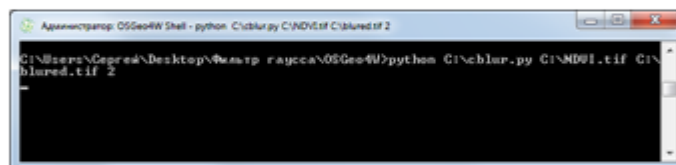
## Блок-схема реализации алгоритма фильтра Гаусса



8

Рисунок А.8 — Слайд презентации № 8

## Интерфейс модуля, реализующего фильтр Гаусса

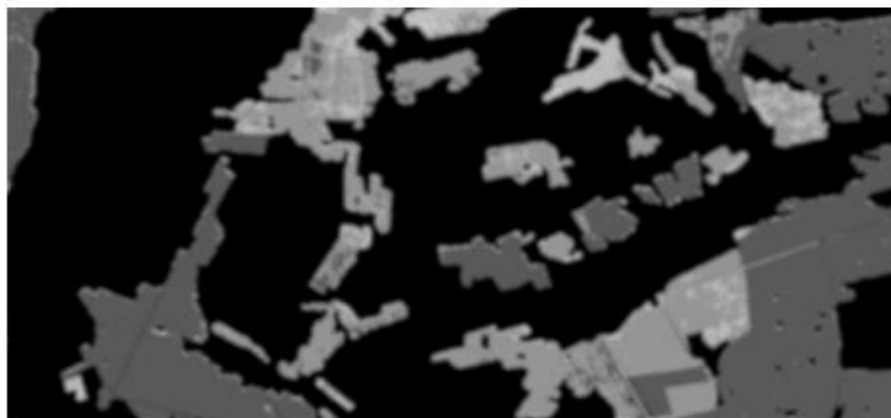


Модуль в OSGeo4W

9

Рисунок А.9 — Слайд презентации № 9

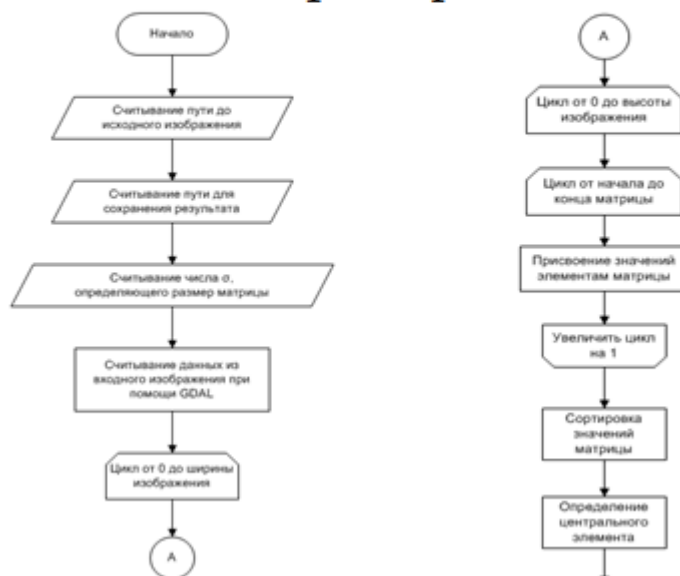
## Результат работы фильтра Гаусса



10

Рисунок А.10 — Слайд презентации № 10

## Блок-схема реализации медианного фильтра



11

Рисунок А.11 — Слайд презентации № 11

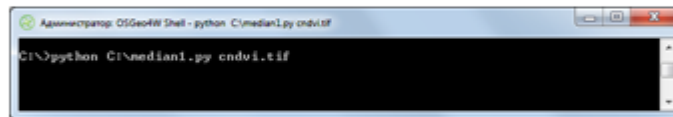
## Блок-схема реализации медианного фильтра



12

Рисунок А.12 — Слайд презентации № 12

## Интерфейс модуля, реализующего медианный фильтр

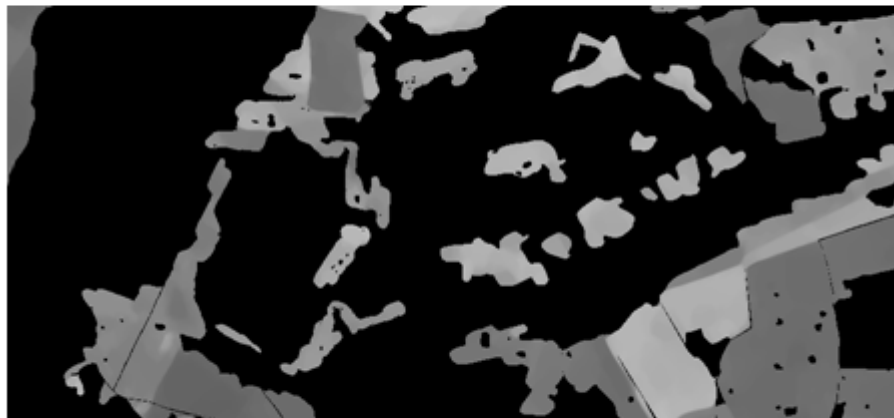


Модуль в OSGeo4W

13

Рисунок А.13 — Слайд презентации № 13

## Результат работы медианного фильтра



14

Рисунок А.14 — Слайд презентации № 14

## Заключение

В рамках поставленной цели были созданы модули, реализующие фильтр Гаусса и медианный фильтр.

Решены все поставленные задачи.

13

Рисунок А.15 — Слайд презентации № 5

# ПРИЛОЖЕНИЕ Г

## Отчет «Антиплагиат»

Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное  
учреждение высшего образования  
«Сибирский федеральный университет»

### НАУЧНАЯ БИБЛИОТЕКА

660049, Красноярск, пр. Свободный, 79/10, тел.(3912) 2-912-820, факс (3912) 2-912-773  
E-mail: bik@sfu-kras.ru

#### ОТЧЕТ

о результатах проверки в системе «АНТИПЛАГИАТ»

Автор: Падалко Сергей Витальевич

Заглавие: Разработка модуля устранения шумов в задачах постсегментации спутниковых изображений

Вид документа: Выпускная квалификационная работа бакалавра

По результатам проверки оригинальный текст составляет 88,49%

Источник	Коллекция / модуль поиска	Ссылка на источник	Доля в отчете	Доля в тексте
Алгоритмические основы раст ровой машинной графики. Уче бное пособие	bibliorossica	<a href="http://www.bibliorossica.com/book.html?&amp;currBookId=12126">http://www.bibliorossica.com/book.html?&amp;currBookId=12126</a>	3,02	3,02
Цифровая обработка изображе	bibliorossica	<a href="http://www.bibliorossica.com/book.html?&amp;currBookId=16218">http://www.bibliorossica.com/book.html?&amp;currBookId=16218</a>	0	0,48
Метод поля направлений в ана лизе и интерпретации диагнос тических изображений : учебн ое пособие	bibliorossica	<a href="http://www.bibliorossica.com/book.html?&amp;currBookId=8984">http://www.bibliorossica.com/book.html?&amp;currBookId=8984</a>	0,01	0,41
Оценивание геометрических п араметров биомедицинских ди агностических изображений : учебное пособие	bibliorossica	<a href="http://www.bibliorossica.com/book.html?&amp;currBookId=8764">http://www.bibliorossica.com/book.html?&amp;currBookId=8764</a>	0	0,27
Цифровые видеоинформацион ные системы (теория и практи	bibliorossica	<a href="http://www.bibliorossica.com/book.html?&amp;currBookId=16219">http://www.bibliorossica.com/book.html?&amp;currBookId=16219</a>	0	0,22
233998	directmedia	<a href="http://biblioclub.ru/index.php?page=book_red&amp;id=233998">http://biblioclub.ru/index.php?page=book_red&amp;id=233998</a>	0	3,02
229578	directmedia	<a href="http://biblioclub.ru/index.php?page=book_red&amp;id=229578">http://biblioclub.ru/index.php?page=book_red&amp;id=229578</a>	1,08	1,08
275218	directmedia	<a href="http://biblioclub.ru/index.php?page=book_red&amp;id=275218">http://biblioclub.ru/index.php?page=book_red&amp;id=275218</a>	0,35	0,55
233465	directmedia	<a href="http://biblioclub.ru/index.php?page=book_red&amp;id=233465">http://biblioclub.ru/index.php?page=book_red&amp;id=233465</a>	0	0,48